

# Trabajo Fin de Máster

## Ingeniería de Telecomunicación

### Definición de un Flujo de Trabajo para el Desarrollo de Modelos Computacionales Personalizados del Cerebro

Autora: Lucía Tejero Torres

Tutor: Luis Javier Reina Tosina

Cotutora: M. Amparo Callejón Leblic

**Dpto. Teoría de la Señal y Comunicaciones**  
**Escuela Técnica Superior de Ingeniería**  
**Universidad de Sevilla**

Sevilla, 2018





Trabajo Fin de Máster  
Ingeniería de Telecomunicación

# **Definición de un Flujo de Trabajo para el Desarrollo de Modelos Computacionales Personalizados del Cerebro**

Autora:

Lucía Tejero Torres

Tutor:

Luis Javier Reina Tosina

Profesor Titular de Universidad

Cotutora:

M. Amparo Callejón Leblic

Dra. Ingeniera de Telecomunicación

Dpto. de Teoría de la Señal y Comunicaciones

Escuela Técnica Superior de Ingeniería

Universidad de Sevilla

Sevilla, 2018



Trabajo Fin de Máster: Definición de un Flujo de Trabajo para el Desarrollo de Modelos Computacionales  
Personalizados del Cerebro

Autora: Lucía Tejero Torres

Tutor: Luis Javier Reina Tosina

Cotutora: M. Amparo Callejón Leblic

El tribunal nombrado para juzgar el Proyecto arriba indicado, compuesto por los siguientes miembros:

Presidente:

Vocales:

Secretario:

Acuerdan otorgarle la calificación de:

Sevilla, 2018

El Secretario del Tribunal



*A mi familia*

*A mis maestros*





# Agradecimientos

---

Este trabajo está dedicado a todas aquellas personas que me han ayudado y apoyado durante este proceso de aprendizaje.

En primer lugar, quiero dar las gracias a mi tutor y profesor del Máster Luis Javier Reina, por haberme transmitido sus conocimientos durante la enseñanza y ofrecerme la oportunidad de realizar este trabajo. A mi cotutora M. Amparo Callejón, a quien le estoy inmensamente agradecida por su ayuda y confianza en mi desde el primer momento.

Agradecer a mis padres, Antonio y M<sup>a</sup> Ángeles, y a mis hermanas Olga, M<sup>a</sup> Ángeles y Julia, por todo el amor que me demuestran cada día que es tan necesario para afrontar cualquier adversidad.

Gracias a Ricardo, por ser mi apoyo y darme fuerzas para seguir en esos días en los que las he necesitado.

También dar las gracias a mis compañeros del Máster por haber hecho más llevaderos estos dos años.

Por último, agradecer a Pablo Mir por ofrecerme la oportunidad de trabajar en el IBIS, y a los compañeros del Grupo de Trastornos del Movimiento, especialmente a Juan Francisco Martín, Antonio Rodríguez, Miguel Ángel Labrador y José Gómez-Feria por ayudarme en la ardua, pero gratificante tarea de la investigación.

*Lucía Tejero Torres*

*Sevilla, 2018*



Con la llegada de la neuroimagen ha sido posible estudiar el cerebro de pacientes vivos con el uso de técnicas no invasivas apoyadas en el análisis de imágenes médicas. Las imágenes de resonancia magnética (MRI, de sus siglas en inglés) se han vuelto muy importantes en la investigación, constituyendo su procesamiento un amplio ámbito de estudio. El procesamiento de las imágenes comprende [1] la obtención de cada tejido, registración de la imagen, reconstrucción de la imagen [2], normalización y suavizado espacial, etc.

Las MRIs son la base para realizar un modelado de la cabeza humana con el fin de analizar la anatomía del individuo y estudiar los flujos de corriente del cerebro, todo ello en relación a la técnica de estimulación de corriente directa transcraneal (tDCS) [3], estimulación magnética transcraneal (TMS) y la electroencefalografía (EEG) [4]. Para realizar estos estudios es necesario tener un modelo computacional de la cabeza del sujeto, que consiste en realizar el procesamiento de las imágenes, reconstruir esas imágenes en volumen y colocar unos electrodos sobre el cuero cabelludo [5].

Existen numerosas herramientas para la elaboración de esos modelos [6]. En este trabajo se ha realizado un estudio de las herramientas más influyentes, haciendo simulaciones de los procesos implicados en la construcción del modelo computacional para analizar diferencias [7], limitaciones o ventajas de unas frente a otras.

Antes de realizar el estudio de las herramientas, fue necesaria una revisión de los formatos utilizados en las imágenes biomédicas, así como un estudio más profundo del formato NIFTI. También se han investigado otras formas de visualizar una MRI, para aclarar conceptos en cuanto a coordenadas vóxel, coordenadas mundo, valores representados en la imagen, orientación de la imagen, etc.

Se realizó un estudio de SPM12 [8], ROAST y SIMNIBS en relación a la segmentación de las imágenes. Con los resultados obtenidos se procedió al mallado superficial realizado con iso2mesh [9], haciendo una comparativa de los métodos binsurface y vol2surf. Y por último se estudiaron las herramientas Metch, de iso2mesh [10], y Mesh2EEG para poder colocar los electrodos sobre la cabeza del sujeto. Tras analizar los resultados obtenidos y su posterior elección de métodos, se ha realizado un flujo de trabajo que contiene todos estos pasos realizados de forma automática en su mayor parte.

Por último, todos los conocimientos adquiridos han sido aplicados para resolver el sistema de posicionamiento del software Brainsight. Este software se creó para desarrollar nuevas herramientas que ayudaran a la investigación de la neurociencia. Actualmente, es el neuronavegador más popular para TMS [11]. Aprovechando el modelo computacional [12] generado a partir del flujo de trabajo realizado, se localizó un punto dentro del sistema de coordenadas de este modelo computacional y se buscó la correspondencia con el sistema de coordenadas de Brainsight. De esta forma se pueden hacer estudios previos en el modelo computacional del sujeto estimulando una zona objetivo y, tras los resultados, estimular exactamente esa misma zona en la cabeza real del paciente.



With the arrival of neuroimaging it has been possible to study the brain of living patients with the use of non-invasive techniques based on medical images, among others. Magnetic resonance imaging (MRI) has become very important in research, its processing being a broad field of study. The processing of the images [1] consists of the obtaining of tissues, image registration, image reconstruction [2], image normalization and the spatial smoothing, etc.

MRIs are the basis for a human head modeling in order to analyze the anatomy of the individual and study the current flows of the brain, all in relation to the transcranial direct current stimulation (tDCS [3]) technique, magnetic transcranial stimulation (TMS) and electroencephalography (EEG [4]). To carry out these studies it is necessary to have a computational model of the subject's head, which consists in carrying out the processing of the images, reconstructing those images in volume and placing electrodes on the scalp [5].

There are numerous tools for the development of these models [6]. In this work, a study of the most influential tools has been made, making simulations of the processes involved in making the computational model to analyze differences [7], limitations or advantages of some against others.

Before carrying out the study of the tools, it was necessary to review the formats used in biomedical imaging, as well as a more in-depth study of the NIFTI format. Other ways to visualize an MRI have also been investigated, to clarify concepts regarding voxel coordinates, world coordinates, values represented in the image, image orientation, etc.

A study of SPM12 [8], ROAST and SIMNIBS was carried out in relation to the segmentation of the images. With the obtained results, we proceeded to the surface meshing carried out with iso2mesh [9], making a comparison of the binsurface and vol2surf methods. And finally, the Metch, iso2mesh [10], and Mesh2EEG tools were studied in order to place the electrodes on the subject's head. After analyzing the results obtained and their subsequent choice of methods, a workflow has been carried out that contains all these steps performed automatically for the most part.

Finally, all the knowledge acquired has been applied to solve the Brainsight software positioning system. This software was created to develop new tools that will help neuroscience research. Currently, it is the most popular neuronavigator for TMS [11]. Taking advantage of the computational model [12] created with the workflow carried out, a point was located within the coordinate system of this computational model and correspondence was sought with the Brainsight coordinate system. In this way, previous studies can be done in the computational model of the subject stimulating a specific area and, after the results, stimulate exactly that same area in the real head of the patient.

|  |             |
|--|-------------|
| <b>Agradecimientos</b>   | <b>ix</b>   |
| <b>Resumen</b>   | <b>xi</b>   |
| <b>Abstract</b>  | <b>xiii</b> |
| <b>Índice</b>  | <b>xiv</b>  |
| <b>Índice de Figuras</b>   | <b>xvi</b>  |
| <b>Notación</b>  | <b>xx</b>   |
| <b>1 Introducción</b>  | <b>1</b>    |
| 1.1. <i>Antecedentes</i>   | 1           |
| 1.2. <i>Objetivos</i>  | 2           |
| 1.3. <i>Estructura de la memoria</i>   | 3           |
| <b>2 Materiales y Métodos</b>  | <b>5</b>    |
| 2.1. <i>Formato DICOM</i>  | 5           |
| 2.2. <i>Formato NIFTI</i>  | 7           |
| 2.3. <i>Otras funciones para representar la MRI</i>                                    | 16          |
| 2.4. <i>Metodología</i>  | 25          |
| <b>3 Resultados</b>  | <b>39</b>   |
| 3.1 <i>Introducción</i>  | 39          |
| 3.2 <i>Resultados de la segmentación realizada con SPM12</i>                           | 39          |
| 3.3 <i>Resultados de la segmentación realizada con ROAST</i>                           | 42          |
| 3.4 <i>Resultados de la segmentación realizada con SIMNIBS</i>                         | 46          |
| 3.5 <i>Resultados del mallado superficial</i>  | 49          |
| 3.6 <i>Resultados del mallado volumétrico con Surf2mesh</i>                            | 53          |
| 3.7 <i>Resultados de la colocación de electrodos</i>                                   | 54          |
| 3.8 <i>Flujo de trabajo para la elaboración de un modelo computacional del cerebro</i> | 55          |
| 3.9 <i>Aplicación al software Brainsight</i>   | 60          |
| <b>4 Conclusiones</b>  | <b>65</b>   |
| <b>Referencias</b>   | <b>67</b>   |



# ÍNDICE DE FIGURAS

---

|  |    |
|--|----|
| Figura 1. Vista sagital de un sujeto con MicroDicom  | 6  |
| Figura 2. Vista transversal de un sujeto con MicroDicom  | 6  |
| Figura 3. Vista coronal de un sujeto con MicroDicom  | 7  |
| Figura 4. Orientación RAS  | 8  |
| Figura 5. Campos de una imagen .nii cargada con <code>load_nii.m</code>                            | 10 |
| Figura 6. Campos de la cabecera, <code>hdr</code> , cargada con <code>load_nii</code>              | 11 |
| Figura 7. Campos dentro de la estructura <code>original</code> , cargada con <code>load_nii</code> | 11 |
| Figura 8. Campos de imagen .nii cargada con <code>load_untouch_nii</code>                          | 12 |
| Figura 9. Campos de la cabecera, <code>hdr</code> , cargados con <code>load_untouch_nii</code>     | 12 |
| Figura 10. Visualización de la imagen T1 con <code>view_nii</code>                                 | 13 |
| Figura 11. Representación de imagen T1 con la herramienta de NIFTI de Matlab en rosa               | 14 |
| Figura 12. Conversión a cualquier orientación  | 15 |
| Figura 13. Visualización, con <code>view_nii</code> , de la imagen T1 reorientada                  | 15 |
| Figura 14. Representación de T1 con <code>imshow</code>  | 16 |
| Figura 15. Representación de T1 con <code>imagesc</code>   | 17 |
| Figura 16. Representación de T1 con <code>image</code>   | 17 |
| Figura 17. Escalado de ejes en <code>imshow</code>   | 18 |
| Figura 18. Sistema de coordenadas intrínseca de la imagen  | 19 |
| Figura 19. Plantilla de MRICron con opción de <code>zoom fit</code>                                | 20 |
| Figura 20. Información de cabecera NIFTI, sección <code>Dimensions</code>                          | 20 |
| Figura 21. Información de cabecera NIFTI, sección <code>Reorient</code>                            | 21 |
| Figura 22. Imagen normalizada sobre plantilla normalizada  | 22 |
| Figura 23. Imagen no normalizada sobre plantilla normalizada                                       | 22 |
| Figura 24. Plantilla normalizada a la izquierda. MRI no normalizada a la derecha                   | 23 |
| Figura 25. Ejemplos de orientaciones posibles en MRICron   | 23 |
| Figura 26. Ejes <code>x</code> e <code>y</code> invertidos.  | 24 |
| Figura 27. T1.nii descargada de la web PPMI y representada con MRICron                             | 24 |
| Figura 28. T2.nii descargada de la web PPMI y representada con MRICron                             | 25 |
| Figura 29. Pantalla de inicio SPM12  | 26 |
| Figura 30. Corregistro y alineación de imágenes  | 27 |
| Figura 31. Display para visualizar las imágenes  | 27 |
| Figura 32. Opciones del Display  | 28 |
| Figura 33. Opción <code>Segment</code> del menú principal  | 29 |
| Figura 34. Batch Editor para <code>Segment</code>  | 29 |



|  |    |
|--|----|
| Figura 35. Esquema de ISO2MESH para creación de mallas volumétricas  | 31 |
| Figura 36. Sistema internacional 10/20 sobre la MRI  | 33 |
| Figura 37. Sistema internacional 10/20 sobre MRI completo  | 33 |
| Figura 38. Sistema 10/05   | 34 |
| Figura 39. Sistema 10/10   | 34 |
| Figura 40. Ubicación de electrodos con la herramienta Metch. Mapeo   | 35 |
| Figura 41. Ubicación de electrodos con la herramienta Metch. Parte trasera de la cabeza  | 35 |
| Figura 42. Ubicación de electrodos con la herramienta Metch. Parte lateral de la cabeza  | 35 |
| Figura 43. Ubicación de electrodos con la herramienta Mesh2EEG. Parte lateral de la cabeza   | 36 |
| Figura 44. Ubicación de electrodos con la herramienta Mesh2EEG. Parte trasera de la cabeza   | 36 |
| Figura 45. Electrodo colocados con ROAST sobre la MRI  | 37 |
| Figura 46. Electrodo y fluido cerebroespinal mallados con ROAST  | 37 |
| Figura 47. Resultados de la segmentación de T2   | 40 |
| Figura 48. Comparativa de segmentación de cráneo con T1 (derecha), y T2 (izquierda)  | 41 |
| Figura 49. Smoothing de los tejidos segmentados  | 42 |
| Figura 50. Segmentación con ROAST usando el template de ROAST, eTPM.nii (izquierda), y usando el template de SPM8, TPM.nii (derecha) | 43 |
| Figura 51. Segmentación del cráneo con ROAST (eTPM), con SPM8 (izquierda) y SPM12 (derecha)  | 44 |
| Figura 52. Segmentación del fluido cerebroespinal con ROAST (eTPM), con SPM8 (izquierda) y SPM12 (derecha)                           | 44 |
| Figura 53. Segmentación de la materia gris con ROAST (eTPM), con SPM8 (izquierda) y SPM12 (derecha)                                  | 44 |
| Figura 54. Segmentación de la materia blanca con ROAST (eTPM), con SPM8 (izquierda) y SPM12 (derecha)                                | 45 |
| Figura 55. Segmentación de la piel con ROAST (eTPM), con SPM8 (izquierda) y SPM12 (derecha)  | 45 |
| Figura 56. Máscaras binarias con ROAST, SPM8 (izquierda), SPM12 (derecha)  | 46 |
| Figura 57. Segmentación de FreeSurfer, para reconstrucción de malla volumétrica  | 47 |
| Figura 58. Segmentación de FreeSurfer de materia blanca y gris   | 47 |
| Figura 59. Segmentación del cerebelo (izquierda) y CSF (derecha) representada con NIFTI toolbox                                      | 48 |
| Figura 60. Segmentación de la materia gris (izquierda) y de la piel (derecha) representada con NIFTI toolbox                         | 48 |
| Figura 61. Segmentación de la materia blanca (izquierda) y cráneo (derecha) representada con NIFTI toolbox                           | 48 |
| Figura 62. Segmentación de ventrículos representada con NIFTI toolbox  | 48 |
| Figura 63. Malla superficial de la piel creada con binsurface a partir de segmentación con SPM8 (izquierda) y SPM12 (derecha)        | 49 |
| Figura 64. Malla superficial del CSF creada con binsurface a partir de segmentación con SPM8 (izquierda) y SPM12 (derecha)           | 49 |
| Figura 65. Malla superficial del cráneo creada con binsurface a partir de segmentación con SPM8 (izquierda) y SPM12 (derecha)        | 49 |
| Figura 66. Malla superficial de la materia gris creada con binsurface a partir de segmentación con SPM8                              |    |

|  |    |
|--|----|
| (izquierda) y SPM12 (derecha)  | 50 |
| Figura 67. Malla superficial de la materia blanca creada con binsurface a partir de segmentación con SPM8 (izquierda) y SPM12 (derecha)      | 50 |
| Figura 68. Malla superficial de la piel (izquierda) y del cerebelo (derecha) creada con binsurface a partir de la segmentación de SIMNIBS    | 50 |
| Figura 69. Malla superficial del CSF (izquierda) y de la materia gris (derecha) creada con binsurface a partir de la segmentación de SIMNIBS | 51 |
| Figura 70. Malla superficial del cráneo (izquierda) y ventrículos (derecha) creada con binsurface a partir de la segmentación de SIMNIBS     | 51 |
| Figura 71. Malla superficial de la materia blanca creada con binsurface a partir de la segmentación de SIMNIBS                               | 51 |
| Figura 72. Malla superficial de la piel con método cgalsurf  | 51 |
| Figura 73 Malla superficial del cráneo con método cgalsurf   | 52 |
| Figura 74. Malla superficial del cráneo con método cgalsurf. Menor tamaño de triángulos  | 52 |
| Figura 75. Mallado del CSF mediante ROAST.   | 53 |
| Figura 76. Fusión de los tejidos segmentados con SIMNIBS   | 53 |
| Figura 77. Malla de tetraedros a partir de los tejidos segmentados con SIMNIBS   | 54 |
| Figura 78. Colocación de electrodos con algoritmo mejorado   | 55 |
| Figura 79. Mensaje en consola de Matlab para introducir la ruta del directorio principal   | 55 |
| Figura 80. Cuadro de diálogo para la selección del directorio principal  | 56 |
| Figura 81. Conversión de archivos DICOM a archivos NIFTI y descompresión de archivos NIFTI   | 56 |
| Figura 82. Reorientación de la imagen T1 (izquierda) y T2 (derecha)  | 57 |
| Figura 83. Creación de máscaras binarias e indicación para seleccionar el directorio donde encuentran  | 57 |
| Figura 84. Cuadro de diálogo para la selección del directorio de las máscaras  | 58 |
| Figura 85. Cuadro de diálogo para seleccionar la máscara de la piel  | 58 |
| Figura 86. Introducción del radio y espesor del cilindro   | 58 |
| Figura 87. Localización del punto de referencia Nasion   | 59 |
| Figura 88. Puntos de referencias insertados  | 59 |
| Figura 89. Selección de electrodos a colocar sobre la cabeza   | 59 |
| Figura 90. Electrodos sobre la malla superficial de triángulos   | 60 |
| Figura 91. Coordenada en vóxel de un punto sobre la malla superficial  | 61 |
| Figura 92. Coordenada en vóxel de un punto sobre la MRI del Brainsight   | 62 |
| Figura 93. Coordenada en milímetros de un punto sobre la MRI del Brainsight  | 62 |
| Figura 94. Interfaz del Brainsight   | 63 |
| Figura 95. Localización de un punto en el espacio real con el software Brainsight  | 63 |



# Notación

---

|       |  |
|-------|--|
| MRI   | Magnetic Resonance Imaging   |
| FMRI  | Functional Magnetic Resonance Imaging  |
| TAC   | Tomografía Axial Computarizada   |
| EEG   | Electroencefalografía  |
| MEG   | Magnetoencefalografía  |
| PET   | Positron Emission Tomography   |
| tDCS  | Transcranial direct-current Stimulation                                      |
| TMS   | Transcranial Magnetic Stimulation  |
| JPEG  | Joint Photographic Experts Group   |
| GIF   | Graphics Interchange Format  |
| BMP   | Bit Mapped Picture   |
| PNG   | Portable Network Graphics  |
| TIFF  | Tagged Image File Format   |
| SPECT | Single Photon Emission Computed Tomography                                   |
| MINC  | McConnell Brain Imaging Centre   |
| MNI   | Montreal Neurological Institute  |
| SPM   | Statistical Parametric Mapping   |
| FSL   | FMRIB Software Library   |
| FMRIB | Functional Magnetic Resonance Imaging of the Brain                           |
| ROAST | Realistic volumetric-Approach to Stimulate Transcranial electric stimulation |
| DICOM | Digital Imaging and Communication in Medicine                                |
| NIFTI | Neuroimaging Informatics Technology Initiative                               |
| CSF   | Cerebral Spinal Fluid  |
| WM    | White Matter   |
| GM    | Gray Matter  |

# 1 INTRODUCCIÓN

---

## 1.1. Antecedentes

El cerebro es el órgano que constituye el sistema nervioso central y el órgano de mayor relevancia en el cuerpo humano. Su estudio es determinante para entender a los individuos. Disciplinas como la filosofía, sociología, psicología o neurología han aportado diferentes puntos de vista acerca del funcionamiento cognitivo de los seres humanos. Sin embargo, han sido las técnicas de la neuroimagen las que han permitido el estudio del cerebro de personas vivas mediante herramientas mínimamente invasivas.

Las técnicas de neuroimagen son capaces de obtener una representación precisa anatómica y/o funcional del cerebro. Han sido muy utilizadas por la neurociencia a partir del descubrimiento de los rayos X. Existen dos tipos de técnicas dentro de la neuroimagen, como son la neuroimagen estructural y funcional. La neuroimagen estructural ofrece una visión estática del cerebro humano [13]. Se utiliza para medir el tamaño total del cerebro y las regiones específicas dentro de él, pudiendo detectar alguna anomalía o efectos de una enfermedad. La neuroimagen funcional ofrece una visión dinámica del funcionamiento del cerebro durante el desempeño de alguna actividad psicológica/ cognitiva.

La tomografía por emisión de positrones [14] o el electroencefalograma (EEG) [15] son técnicas que pertenecen a la neuroimagen funcional. La tomografía por emisión de positrones (PET) es una técnica invasiva. Permite identificar si se ha producido una disminución del número de neuronas inyectando en el sujeto una sustancia radioactiva. Con el electroencefalograma (EEG) se realiza un estudio de la actividad eléctrica cerebral a través de unos electrodos que se colocan sobre el cuero cabelludo de acuerdo a un estándar internacional. Es una técnica no invasiva pero poco precisa en cuanto a conocer exactamente el lugar por donde se ha producido la actividad neuronal. Por ello, debe ser complementada por técnicas de neuroimagen. Permite monitorizar la actividad eléctrica del cerebro durante un estímulo.

La tomografía axial computarizada consiste en enviar rayos X al cerebro, a través del cráneo, obteniendo múltiples imágenes del mismo. La finalidad de esta técnica es la detección de anomalías. La imagen de resonancia magnética (MRI) [16] o la tomografía axial computarizada (TAC) son dos técnicas que pertenecen a la parte estructural de la neuroimagen. La imagen de resonancia magnética sirve para estudiar la anatomía del cerebro. Consiste en la emisión de un campo magnético sobre el sujeto y registrar la información a través de un ordenador, que la transforma en una imagen. Hay diferentes tipos de MRI como por ejemplo T1 y T2. Las imágenes T1 pueden considerarse como un mapa de la energía del protón dentro de los tejidos grasos del cuerpo que son la grasa subcutánea y médula ósea de las vertebras. El líquido cefalorraquídeo no contiene grasa, por lo que aparece en negro en las imágenes T1. Por otro lado, las imágenes T2 son un mapa de la energía del protón dentro de los tejidos grasos y de los tejidos que contienen agua del cuerpo. Cualquier zona que sea brillante en las imágenes T2 pero oscura en las imágenes T1 es un tejido formado por agua.

Por otra parte, el análisis de la imagen de resonancia magnética se ha demostrado muy importante tanto en investigación como en estudios clínicos destinados al desarrollo, la función y la patología del cerebro humano. Y sobre todo en estudios relacionados con terapias orientadas a la resolución de problemas neurológicos

mediante técnicas no invasivas, entre las que destacan los métodos de estimulación neuronal. Éstas se basan en la aplicación de energía eléctrica, electromagnética u óptica, y tienen como objetivo la activación, inhibición y/o regulación de la actividad neuronal dependiendo de la patología a tratar. Entre las técnicas que utilizan fuentes electromagnéticas, por su carácter no invasivo destacan la estimulación magnética (TMS) y eléctrica (tES) transcraneal [17] [18]. En ambos casos, la técnica se aplica disponiendo una bobina o unos electrodos [19] en la parte externa de la cabeza del paciente, a través de los cuales se induce una corriente eléctrica sobre los tejidos cerebrales. Los investigadores usan esta técnica para conocer cómo se relaciona la actividad cerebral con los procesos cognitivos y el comportamiento del paciente. También se utiliza para estudiar enfermedades como Parkinson, depresión o epilepsia. A pesar de los avances logrados con estas técnicas, se desconocen los mecanismos bioeléctricos que subyacen al proceso de neuroestimulación, y se evidencia una falta de metodología para el establecimiento de los parámetros de estimulación que, a menudo, se fijan en base a la experiencia del médico/neurofisiólogo y los estándares de seguridad.

Los avances en modelado computacional han permitido desentrañar algunos de los aspectos relacionados con el acoplo de campos electromagnéticos y los tejidos biológicos, contribuyendo a que se pueda cubrir el salto entre la aplicación de las técnicas de neuroestimulación y una planificación de la intervención. Estos modelos proporcionan conocimiento sobre los flujos de corriente en los distintos tejidos cerebrales, y por tanto se apoyan en una representación realista de la anatomía de la cabeza sobre la que se va a intervenir, que se deriva, con frecuencia, de la reconstrucción tridimensional a partir de imágenes MRI. Así, las imágenes de resonancia magnética constituyen una base fundamental para realizar un modelado computacional de la cabeza humana. Existen numerosas herramientas destinadas al tratamiento de las imágenes, y entre sus funciones se incluye la conversión de formato de imagen, registración de la imagen, enmascaramiento de la imagen dentro del cerebro, normalización espacial del sistema de coordenadas, suavizado espacial, segmentación, reconstrucción de la imagen, simulación, visualización de resultados, etc. Algunas de estas herramientas son SPM [20], FreeSurfer [21], FSL, iso2mesh, ROAST, entre otras [22]. No todas estas herramientas sirven para realizar el modelo completo, cada una tiene sus ventajas y sus limitaciones. Dada la diversidad de herramientas, la calidad de los modelos computacionales es fuertemente dependiente de los métodos que utilicen y su secuenciación, echándose en falta una metodología que ayude a seleccionar los procesos necesarios para poder derivar modelos computacionales de utilidad en investigación y terapia. Por este motivo, en este Proyecto se propone un flujo de trabajo que tiene por objeto realizar de forma metódica el proceso de obtener un modelo computacional del cerebro a partir de la MRI de una persona.

## 1.2. Objetivos

Dadas las técnicas experimentales empleadas para estudiar la anatomía del cerebro, como tES, es necesario establecer un procedimiento más metódico que permita un estudio objetivo del cerebro. Para ello, se necesita un modelo computacional del cerebro a partir de MRIs. Esto conlleva a un tratamiento de las imágenes médicas, como segmentación [23], mallado [24] y colocación de electrodos. También influyen los formatos de las imágenes, el sistema de coordenadas, las plantillas utilizadas para hacer la segmentación, etc. Son muchos los detalles que se deben tener en cuenta para el procesamiento de una imagen y muchas las herramientas software que abordan este proceso de diferente forma.

El objetivo de este trabajo fin de máster es elaborar un flujo de trabajo que realice el procedimiento del tratamiento de la imagen de resonancia magnética hasta obtener un modelo computacional del cerebro. Para ello, se estudiarán diferentes herramientas y así comprender de forma rigurosa el proceso que hay que seguir, y las ventajas e inconvenientes de cada una de ellas.

Para lograr este objetivo fundamental, se deben cumplir los siguientes subobjetivos:

1. Estudio de los diferentes formatos de imágenes biomédicas más utilizados y herramientas para la visualización de las mismas con el fin de comprender la imagen representada.
2. Análisis de los resultados de segmentación de una misma MRI con diferentes softwares.
3. Análisis del mallado con diferentes softwares y métodos, de las segmentaciones realizadas previamente.
4. Elección de herramientas para la colocación de los electrodos sobre la cabeza del sujeto.

### 1.3. Estructura de la memoria

En este subapartado se definen brevemente los contenidos que aborda la memoria, dividiéndose en cinco capítulos, que se describen a continuación.

1. Introducción. En este capítulo se describen algunos de los métodos para obtener imágenes biomédicas. Además, se describe brevemente el procedimiento para crear un modelo computacional del cerebro y se expone la necesidad de elaborar este trabajo junto con los objetivos del mismo.
2. Materiales y métodos. Se definen los formatos de las MRIs utilizados, así como las diferentes herramientas para su visualización. Se describen los métodos estudiados tanto para la segmentación y mallado de las MRIs, como para la colocación de los electrodos.
3. Resultados. En este apartado se muestran los resultados obtenidos con los métodos anteriores en cada una de las partes del proceso. Luego se realiza la elección de entre esos métodos para elaborar un flujo de trabajo que comprenda cada uno de los pasos explicados anteriormente, incluyendo una serie de mejoras. Por último, se aplican todos los conocimientos adquiridos a un software actualmente en funcionamiento dedicado a la estimulación magnética transcraneal.
4. Conclusiones. En este apartado se realiza una comparativa de los diferentes métodos utilizados, explicando sus ventajas, inconvenientes y limitaciones, proponiendo futuras mejoras.
5. Referencias. Se expone la bibliografía a la que se ha recurrido para realizar este trabajo.





## 2 MATERIALES Y MÉTODOS

---

**E**n este capítulo se pretende comprender diferentes herramientas para la visualización de las imágenes de resonancia magnética y los posibles formatos en los que se pueden presentar dichas imágenes. Además, se describen los diferentes métodos utilizados para la segmentación y mallado de la MRI y la colocación de electrodos sobre la cabeza del individuo.

En primer lugar, se realiza una definición de los formatos DICOM y NIFTI, pero es de este último sobre el que se profundiza más. Se realiza un estudio acerca de este formato con el uso de la herramienta NIFTI de Matlab, señalando diferencias con el formato DICOM, describiendo las coordenadas en las que se encuentra la imagen, la orientación que toma por defecto, así como la descripción de algunos campos de este formato.

En segundo lugar, se muestran otras formas de representar las MRIs utilizando funciones pertenientes a Matlab, como son `imshow`, `imagesc` o `image`. Describiendo la forma de ejecutar los comandos para su correcta visualización. También se presenta otra herramienta distinta para mostrar la imagen de resonancia magnética, llamada MRICron [25], la cual es de código abierto y dispone de otras funcionalidades además de la visualización de las MRIs.

Por último, se expone la metodología seguida para la elaboración del flujo de trabajo de un modelo computacional, explicando cada uno de los procesos de forma detallada.

### 2.1. Formato DICOM

DICOM (Digital Imaging and Communications in Medicine) es un estándar internacional para transmitir, almacenar, recuperar, imprimir, procesar y representar información de imágenes médicas, las cuales hace interoperables. Se desarrolla y mantiene activamente para cumplir con las tecnologías y necesidades de las imágenes médicas. Está disponible para descargarse y usar gratuitamente.

El formato DICOM surge porque los formatos existentes de imagen anteriores como TIFF (Tagged Image File Format), JPEG (Joint Photographic Experts Group) o GIF (Graphics Interchange Format) solo contienen la imagen médica, sin los datos necesarios del paciente. La solución fue diseñar un nuevo formato de archivos que especificara los datos primordiales de un paciente en la imagen. Además, también era necesaria la comunicación entre equipos heterogéneos con un protocolo de trabajo común para todos los equipos. Así, el estándar DICOM [26] describe un lenguaje común a distintos sistemas médicos. De esta forma las imágenes vienen acompañadas de mediciones, cálculos e información descriptiva relevante para diagnósticos. Utiliza archivos con extensión `.dcm`.

Las comunicaciones DICOM se adaptan al estándar OSI para el intercambio de información. Define interfaces de conexión entre equipos de diferentes categorías como: equipos de adquisición de imagen, archivos de imágenes, estaciones de visualización, dispositivos de impresión, entre otros.

El formato de datos de DICOM agrupa la información en conjuntos de datos. Por ejemplo, un archivo de una imagen de rayos X contiene el identificador del paciente (ID) dentro del archivo, por lo que es imposible separar esta información de la imagen por error. Este tipo de formato no tiene una cabecera como tal, sino que contiene numerosos atributos, como por ejemplo el nombre, el ID, incluso los datos de píxeles. Cabe la posibilidad de que un objeto DICOM contenga solo un atributo que sean los datos de píxeles. Sin embargo, un atributo puede contener múltiples tramas, lo que permite el almacenamiento de bucles de otros datos multi-tramas. Los datos de píxeles se pueden comprimir usando una gran variedad de estándares, incluyendo JPEG.

Para promover la visualización de imágenes idénticas en escala de grises en diferentes monitores, el comité DICOM desarrolló una tabla de búsqueda para mostrar digitalmente los valores de píxeles asignados. Para usar la función de visualización del estándar en escala de grises DICOM (GSDF), las imágenes deben ser visualizadas en un dispositivo que tenga esta tabla de búsquedas o en dispositivos que hayan sido calibrados para esta curva GSDF.

Una imagen de resonancia magnética, por ejemplo, de un cerebro, en formato DICOM consiste en un conjunto de carpetas que contienen capturas del cerebro. Cada carpeta contiene imágenes de una determinada vista (sagital, coronal, transversal). Para la representación de estas imágenes se ha utilizado un software denominado MicroDicom [27]. Este programa es un visor de archivos DICOM gratuito y desarrollado en Windows que permite modificar y conservar las propiedades del archivo original. Se puede realizar listas de pacientes, realizar mediciones y anotaciones, cargar imágenes únicamente arrastrándolas a la interfaz o a través de la barra de herramientas. También permite generar archivos de vídeo, convertir imágenes de formato JPEG, BMP, GIF, TIFF o PNG a formato DICOM. Es totalmente gratuito y disponible para Windows. Las figuras 1-3 muestran imágenes de distintas vistas de la cabeza de un sujeto visualizada con este software.

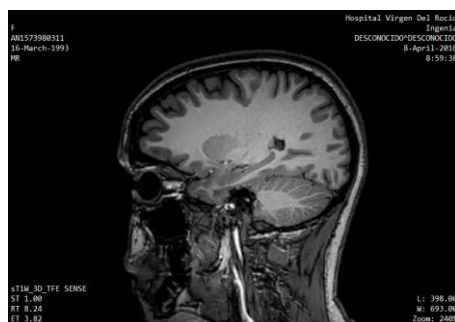


Figura 1. Vista sagital de un sujeto con MicroDicom



Figura 2. Vista transversal de un sujeto con MicroDicom

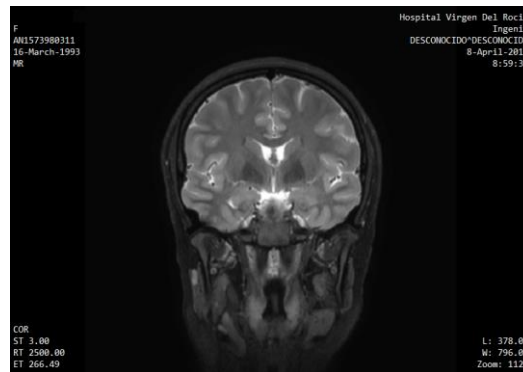


Figura 3. Vista coronal de un sujeto con MicroDicom

## 2.2. Formato NIFTI

El formato NIFTI (Neuroimaging Informatics Technology Initiative) fue desarrollado para facilitar la interoperabilidad multiplataforma y entre programas, así como para solventar algunos problemas que tenía el formato ANALYZE 7.5 [28], como la falta de información en cuanto a la orientación en el espacio. Esto obligó a los nuevos paquetes de software a incluir un archivo que describía la orientación. El nuevo formato fue definido por el Grupo de Trabajo de Formato de Datos (DFWG) de los Institutos Nacionales de Salud de EEUU (NIH) en los días 31 de marzo y 2 de septiembre de 2003 [29].

NIFTI [30] ha estandarizado la representación de imágenes en tres dimensiones del cerebro mientras que DICOM solo representa un corte del cerebro. Puede manejar los datos usando dos archivos, .hdr y .img. La extensión .hdr almacena la metainformación y la extensión .img almacena los datos reales, sin información del paciente, a diferencia del formato DICOM, que almacena muchos campos, información de salud protegida y metadatos relacionados con el hospital.

A diferencia del formato DICOM, NIFTI permite el almacenamiento de varios ficheros en el mismo directorio. En cambio, DICOM tiene una estructura de datos más compleja, almacenando por sujeto diferentes carpetas. La extensión de los archivos NIFTI es .nii, y también pueden comprimirse usando la extensión .nii.gz. Por otro lado, para que el formato NIFTI mantuviera la compatibilidad con el formato ANALYZE no se cambió el tamaño de la cabecera, aunque algunos campos fueron inutilizados, preservados, ignorados o sobrescritos. Dispone de siete dimensiones: las tres primeras definen el espacio  $(x, y, z)$ , mientras que la cuarta dimensión define el tiempo  $t$ . Las demás dimensiones, desde la quinta a la séptima, se destinan a otros usos.

La mejora más notable del formato NIFTI sobre el formato ANALYZE es la habilidad de almacenar la orientación de la información. Asume que las coordenadas del vóxel se refieren al centro de cada vóxel, y no a cualquier esquina. Cada vóxel se corresponde con una coordenada  $(i, j, k)$  que puede no estar orientada correctamente. Sin embargo, el toolbox de NIFTI transforma esta matriz [31] en unas coordenadas espaciales  $(x, y, z)$  para que siempre estén orientadas en RAS. RAS es un tipo de orientación en la que los ejes  $(x, y, z)$  se disponen de la siguiente manera: la dirección del eje  $x$  es hacia la oreja derecha de la cabeza del sujeto, la dirección del eje  $y$  es hacia la parte anterior de la cabeza del sujeto y la dirección del eje  $z$  es hacia la parte superior de la cabeza. (véase la figura 4). La dirección exacta de estos ejes con respecto al sujeto depende de las variables llamadas `qform_code` (Método2) y `sform_code` (Método 3).

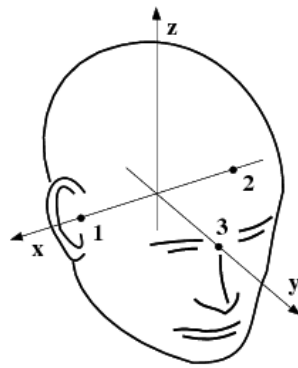


Figura 4. Orientación RAS

Esta transformación se basa únicamente en una inversión y rotación ortogonal para que cada coordenada de los vóxeles  $(i, j, k)$  se alineen con el sistema de coordenadas espacial  $(x, y, z)$  respectivamente. De esta forma, la relación entre los vóxeles y las coordenadas espaciales es simplemente un factor de escala y depende del valor que toman las variables `qform_code` y `sform_code`. Hay tres métodos diferentes para mapear las coordenadas  $(x, y, z)$  a las coordenadas del vóxel  $(i, j, k)$ . Estas coordenadas tienen los siguientes rangos válidos:

$$\begin{aligned} i &= 0 \dots \dim[1] - 1 \\ j &= 0 \dots \dim[2] - 1, \text{ si } \dim[0] \geq 2 \\ k &= 0 \dots \dim[3] - 1, \text{ si } \dim[0] \geq 3 \end{aligned}$$

#### Método 1

Se trata de un simple escalado por el tamaño del vóxel. Este método se usa cuando las variables `sform_code` y `qform_code` son cero:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} i \\ j \\ k \end{bmatrix} \otimes \begin{bmatrix} \text{pixdim}[1] \\ \text{pixdim}[2] \\ \text{pixdim}[3] \end{bmatrix}$$

donde  $\otimes$  es el producto de Hadamard.  $\text{pixdim}[i]$  es el ancho del vóxel a lo largo de la dimensión  $i$ , donde  $i=1 \dots \dim[0]$ . No se adjunta ninguna orientación espacial a las coordenadas  $(x, y, z)$ . Este método está presente por compatibilidad con archivos en formato ANALYZE 7.5.

#### Método 2

Se usa cuando `qform_code` es mayor que cero, que es el caso normal. Muestra los datos en su plantilla original. Las coordenadas  $(x, y, z)$  están dadas por los valores de `pixdim`, por una matriz de rotación y traslación. Este método está destinado a representar coordenadas que están en la cabecera [32] de la imagen:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \mathcal{R} \begin{bmatrix} i \\ j \\ k \end{bmatrix} \otimes \begin{bmatrix} \text{pixdim}[1] \\ \text{pixdim}[2] \\ \text{pixdim}[3] \end{bmatrix} + \begin{bmatrix} \text{qoffset}_x \\ \text{qoffset}_y \\ \text{qoffset}_z \end{bmatrix}$$

El centro del vóxel  $(i, j, k) = (0, 0, 0)$  se corresponde con  $(x, y, z) = (qoffset\_x, qoffset\_y, qoffset\_z)$ . La matriz de rotación  $\mathcal{R}$  se calcula a partir de los parámetros  $quatern\_b$ ,  $quatern\_c$  y  $quatern\_d$ , que se corresponden con las letras  $b$ ,  $c$  y  $d$ , respectivamente. La letra  $a$  se define desde las otras. El factor de escala  $q$  puede tomar los valores 1 o -1.

$$\mathcal{R} = \begin{bmatrix} a^2 + b^2 - c^2 - d^2 & 2(bc - ad) & 2(bd + ac) \\ 2(bc + ad) & a^2 + c^2 - b^2 - d^2 & 2(cd - ab) \\ 2(bd - ac) & 2(cd + ab) & a^2 + d^2 - b^2 - c^2 \end{bmatrix}$$

$$a = \sqrt{1 - b^2 - c^2 - d^2}$$

Si se supone que  $(q, p, r)$  es una unidad de tres vectores, entonces la rotación del ángulo  $h$  sobre esa dirección se representa como:

$$[a, b, c, d] = \cos\left(\frac{h}{2}\right), p * \sin\left(\frac{h}{2}\right), q * \sin\left(\frac{h}{2}\right), r * \sin\left(\frac{h}{2}\right),$$

siendo  $a > 0$ , o lo que es lo mismo  $h$  debe estar comprendido entre  $(-\pi, \pi)$ .

### Método 3

Este método se usa cuando la variable `sform_code` es mayor que cero. Proporciona las ubicaciones de los vóxeles en algún espacio estándar, indicado por el valor de `sform_code`. Se basa en una matriz afín almacenada en la cabecera en los campos `srow_x`, `srow_y`, `srow_z` para mapear los vóxeles a las coordenadas mundo.

$$\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} srow\_x[0] & srow\_x[1] & srow\_x[2] & srow\_x[3] \\ srow\_y[0] & srow\_y[1] & srow\_y[2] & srow\_y[3] \\ srow\_z[0] & srow\_z[1] & srow\_z[2] & srow\_z[3] \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} i \\ j \\ k \\ 1 \end{bmatrix}$$

Otro parámetro importante dentro del formato NIFTI es `magic_string`, que puede tomar dos valores: si es igual a `ni1` significa que los datos de la imagen se almacenan en el archivo `.img` correspondiente al archivo de encabezado; si es igual a `n+1` significa que los datos de la imagen se almacenan en el mismo archivo que los datos de cabecera, es decir, la imagen ha sido cargada como `.nii`.

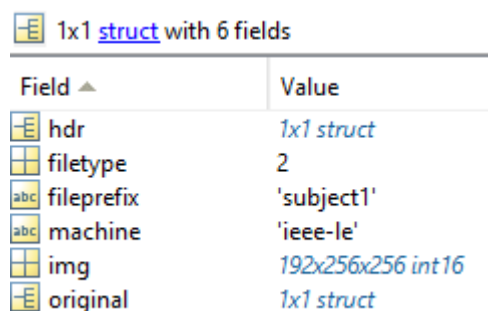
Para entender mejor este formato se ha usado un toolbox de Matlab dedicado al formato NIFTI [33], con el que se han representado las MRIs. Para poder visualizar una MRI usando este toolbox, la imagen debe ser cargada con la función `load_nii`, la cual crea una estructura que contiene varias variables y estructuras. Incluye una

cabecera que, a su vez, almacena varias estructuras donde se guardan otras variables como las explicadas anteriormente en el apartado de “Formato NIFTI”. También almacena la imagen en una matriz de tres dimensiones llamada `img`, y una estructura original con los datos de la imagen antes de realizarle cualquier transformación. Esta función aplica la transformación afín a la imagen, orientándola en RAS. Hay tres posibles situaciones que se pueden encontrar al usar el comando `load_nii('filename.nii')`:

1. Si el fichero `nii.filetype` es cero, significa que la imagen puede tener cualquier orientación debido a que el dato está en formato ANALIZE 7.5.
2. Si el fichero `nii.filetype` no es cero y los campos `nii.hdr.hist.rot_orient` y `nii.hdr.hist.flip_orient` están vacíos, significa que la matriz afín para realizar la re-orientación era una matriz diagonal con enteros positivos, y los datos están almacenados en orientación RAS.
3. Si el fichero `nii.filetype` no es cero y los campos `nii.hdr.hist.rot_orient` y `nii.hdr.hist.flip_orient` no están vacíos, significa que la matriz afín no era una matriz diagonal de enteros positivos y se ha realizado la orientación correspondiente para que la matriz `img` del struct tenga orientación RAS. De esta forma se puede representar la imagen con la función `view_nii`.

```
%Ejemplo: como cargar y visualizar una imagen nifti.
T1_nii=load_nii('T1.nii');
view_nii(T1_nii);
```

La figura 5 muestra los campos que contiene un archivo `.nii` cuando es cargado con la función `load_nii.m` del toolbox de NIFTI [33], donde se pueden observar algunos campos de los indicados anteriormente.



| Field      | Value             |
|------------|-------------------|
| hdr        | 1x1 struct        |
| filetype   | 2                 |
| fileprefix | 'subject1'        |
| machine    | 'ieee-le'         |
| img        | 192x256x256 int16 |
| original   | 1x1 struct        |

Figura 5. Campos de una imagen `.nii` cargada con `load_nii.m`

Se observa que el campo `filetype` no es cero, y si además se observan los campos `rot_orient` y `flip_orient` (figura 6), se puede comprobar que están vacíos. Por tanto, el archivo se encuentra en la situación del caso 2. Los datos están almacenados con orientación RAS. Que los campos `qform_code` y `sform_code` estén a cero (véase la figura 6) significa que la transformación indicada dentro del campo `original` ha sido realizada correctamente.

1x1 struct with 18 fields

| Field ▲     | Value                     |
|-------------|---------------------------|
| descrip     | ''                        |
| aux_file    | ''                        |
| qform_code  | 0                         |
| sform_code  | 0                         |
| quatern_b   | 0                         |
| quatern_c   | 0                         |
| quatern_d   | 0                         |
| qoffset_x   | -95.5614                  |
| qoffset_y   | -98.3899                  |
| qoffset_z   | -92.9410                  |
| srow_x      | [1 0 0 -95.5614]          |
| srow_y      | [0 0.9883 0 -98.3899]     |
| srow_z      | [0 0 0.9883 -92.9410]     |
| intent_name | ''                        |
| magic       | 'n+1'                     |
| originator  | [96.5614 100.5566 95....] |
| rot_orient  | []                        |
| flip_orient | []                        |

Figura 6. Campos de la cabecera, hdr, cargada con load\_nii

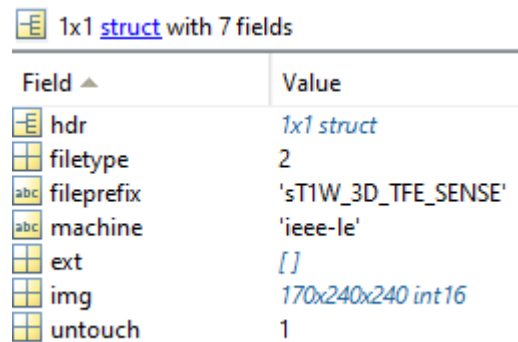
Dentro del campo `original`, que a su vez es otra estructura, se encuentran todos los datos originales de la imagen, sin ningún tipo de transformación. La figura 7 muestra los campos `sform_code` y `qform_code` con valor distinto de cero, concretamente 1. Esto significa que se ha usado el método 3 para hacer la transformación de vóxel  $(i, j, k)$  a coordenadas  $(x, y, z)$ , usando el sistema de referencia de la máquina. El campo `magic` tiene el valor de `n+1`, que quiere decir que el archivo que se ha cargado tiene extensión `.nii`.

1x1 struct with 16 fields

| Field ▲     | Value                 |
|-------------|-----------------------|
| descrip     | ''                    |
| aux_file    | ''                    |
| qform_code  | 1                     |
| sform_code  | 1                     |
| quatern_b   | 0                     |
| quatern_c   | 0                     |
| quatern_d   | 0                     |
| qoffset_x   | -95.5614              |
| qoffset_y   | -98.3899              |
| qoffset_z   | -92.9410              |
| srow_x      | [1 0 0 -95.5614]      |
| srow_y      | [0 0.9883 0 -98.3899] |
| srow_z      | [0 0 0.9883 -92.9410] |
| intent_name | ''                    |
| magic       | 'n+1'                 |
| originator  | [256 0 0 0 0]         |

Figura 7. Campos dentro de la estructura `original`, cargada con load\_nii

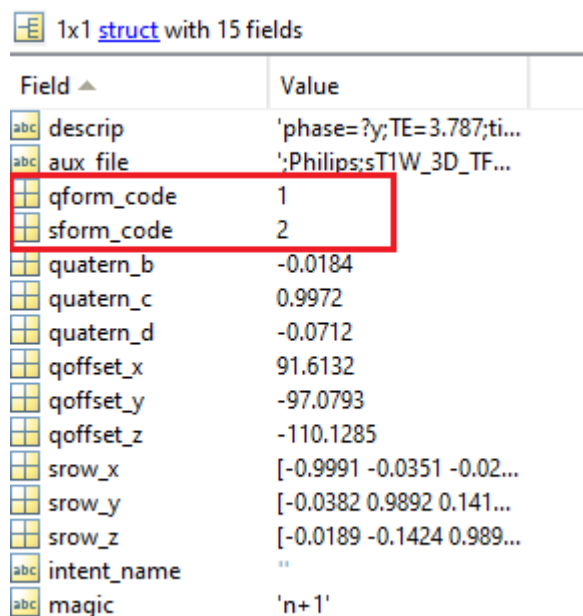
Puede ocurrir que la imagen `.nii` no se pueda cargar con la función `load_nii.m`. Esto sucede cuando la matriz de rotación que se encuentra dentro de este archivo no es ortogonal. En este caso, se debe usar la función `load_untouch_nii.m` para cargar la imagen. Esta función carga la cabecera de la imagen sin realizarle ninguna transformación. La figura 8 muestra los campos que tiene una MRI en formato NIFTI que ha sido cargada con dicha función.



| Field      | Value               |
|------------|---------------------|
| hdr        | 1x1 struct          |
| filetype   | 2                   |
| fileprefix | 'sT1W_3D_TFE_SENSE' |
| machine    | 'ieee-le'           |
| ext        | []                  |
| img        | 170x240x240 int16   |
| untouch    | 1                   |

Figura 8. Campos de imagen `.nii` cargada con `load_untouch_nii`

De nuevo, se observa que el campo `filetype` es distinto de cero, pero cabe destacar que no hay campo `original`, ya que no se ha realizado ninguna transformación, y en cambio existe un nuevo campo, `untouch`, que indica que el archivo ha sido cargado sin alterar la cabecera de la imagen. Al entrar dentro de la cabecera, se observa que el valor de los campos `sform_code` y `qform_code` no son cero, a diferencia del caso anterior. Esto es debido a que no se ha realizado ninguna transformación a la MRI.



| Field       | Value                     |
|-------------|---------------------------|
| descrip     | 'phase=?;TE=3.787;ti...   |
| aux file    | ':Philips;sT1W_3D_TF...   |
| qform_code  | 1                         |
| sform_code  | 2                         |
| quatern_b   | -0.0184                   |
| quatern_c   | 0.9972                    |
| quatern_d   | -0.0712                   |
| qoffset_x   | 91.6132                   |
| qoffset_y   | -97.0793                  |
| qoffset_z   | -110.1285                 |
| srow_x      | [-0.9991 -0.0351 -0.02... |
| srow_y      | [-0.0382 0.9892 0.141...  |
| srow_z      | [-0.0189 -0.1424 0.989... |
| intent_name | ''                        |
| magic       | 'n+1'                     |

Figura 9. Campos de la cabecera, `hdr`, cargados con `load_untouch_nii`

Para poder visualizar la MRI con el toolbox de NIFTI se utiliza la función `view_nii`. Para poder usar esta función, antes debe cargarse la imagen con la función `load_nii` y no con la función `load_untouch_nii`, ya que el archivo debe tener su matriz afin ortogonal. Por tanto, si se quiere visualizar una imagen `.nii` y no



ha podido ser cargada con la función `load_nii`, hay que realizar alguna transformación afín. Para ello, se utiliza la función `reslice_nii` del toolbox de NIFTI. El archivo NIFTI resultante de esta función siempre estará en orientación RAS.

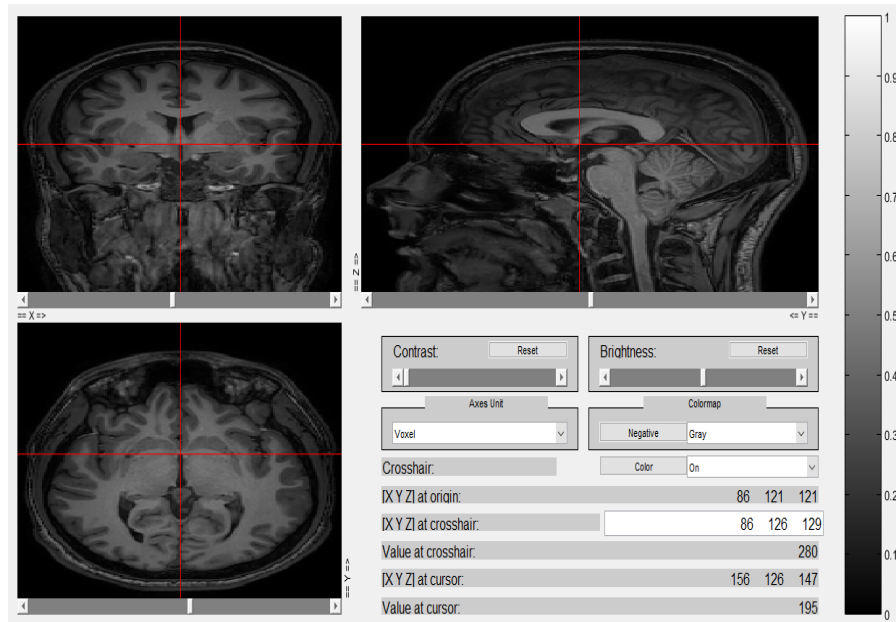


Figura 10. Visualización de la imagen T1 con `view_nii`

La interfaz de esta herramienta (véase la figura 10) presenta un menú en el que se pueden hacer diferentes modificaciones y explorar la imagen:

- Las opciones `Contrast` y `Brightness` sirven para modificar la luminosidad de la imagen.
- `Axes Unit` permite cambiar la unidad de medida, que puede ser vóxel o milímetro.
- La opción `Colormap` se utiliza para cambiar el color de la imagen, cuyo valor por defecto es gris. En la figura 11 se puede observar que los niveles de intensidad de la imagen se representan en una escala de 0 a 1, siendo 0 el color negro y 1 el color blanco. Si se eligiera otro color, también habría una escala de 0 a 1 y se le asignaría a cada valor un color.
- `Color` permite activar o no los ejes de color rojo (`crosshair`) que se ven en la imagen.
- `[X Y Z] at origin` es la posición donde el `crosshair` se sitúa inicialmente.
- `[X Y Z] at crosshair` es la posición fija donde se encuentra el `crosshair`.
- `Value at crosshair` es el valor del vóxel indicado por la posición fija del `crosshair`.
- `[X Y Z] at cursor` es la posición donde el cursor del ratón está situado en ese momento.
- `Value at cursor` es el valor del vóxel indicado por la posición del cursor.

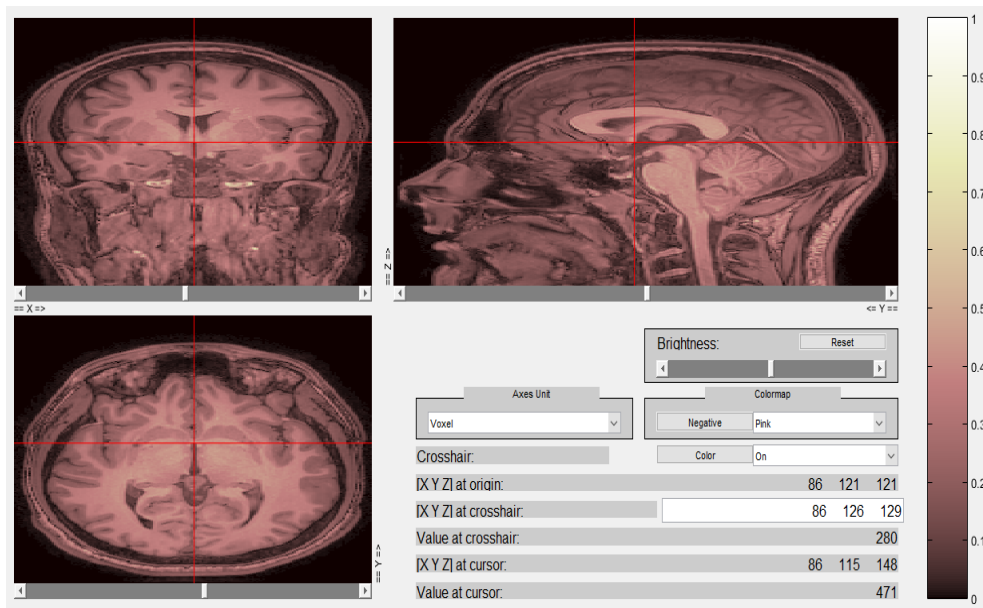


Figura 11. Representación de imagen T1 con la herramienta de NIFTI de Matlab en rosa

La imagen queda almacenada en una matriz `img` de tres dimensiones que contiene los valores de intensidad [34]. Los niveles más altos de intensidad se corresponden con los números más altos de la matriz. Cabe destacar que esas coordenadas  $X, Y, Z$ , cuando la unidad es el vóxel (antes nombrado como  $i, j, k$ ), y el valor del crosshair se corresponden con las posiciones de la matriz `img` y la intensidad en esa posición, respectivamente. Esto se puede comprobar accediendo a esa matriz desde la consola de Matlab. Si se realiza la conversión de unidades de vóxel a milímetros, las coordenadas  $X, Y, Z$  son las llamadas coordenadas mundo.

```
T1_nii.img(86,126,129)
ans =
```

Aunque la orientación de la imagen por defecto es RAS, se puede cambiar a la orientación que se necesite con la función `rri_orient`, cargando primero la MRI con `load_nii`, y pasándole como argumento de entrada el nombre utilizado para cargar la imagen.

```
t1_orient=load_nii('T1.nii')
```

Se abrirá un cuadro de diálogo como el que se muestra en la figura 12. Para modificar la orientación de la imagen, basta con elegir el sentido de los ejes X, Y, Z y pulsar en Done. Se ha elegido la orientación LPS (+X: izquierda, +Y posterior, +Z superior) como ejemplo para representar la imagen (véase la figura 13).

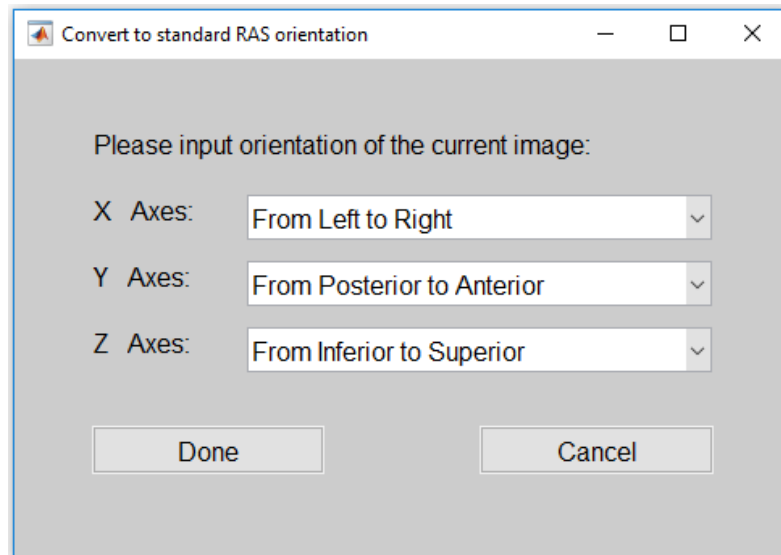


Figura 12. Conversión a cualquier orientación

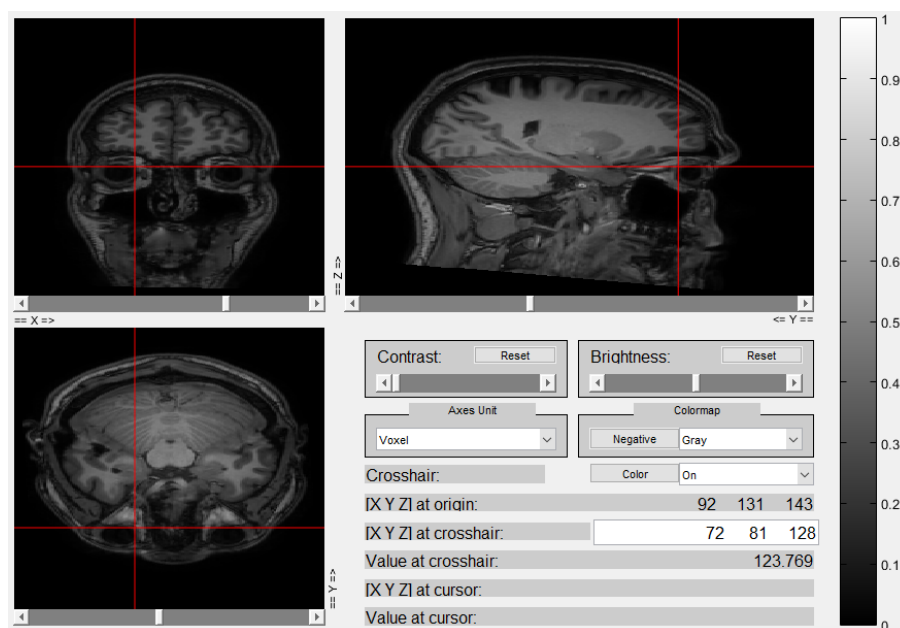


Figura 13. Visualización, con `view_nii`, de la imagen T1 reorientada

Existen otras funciones como `make_nii`, `make_ana`, `save_nii` y `save_untouch_nii` para crear archivos y guardarlos. Las funciones `make_nii` y `make_ana` sirven para crear un archivo `.nii` con su cabecera, tal y como se descrito antes. La diferencia entre ellas es que la primera crea estructuras NIFTI y solo se pueden guardar con la función `save_nii`, mientras que la segunda crea estructuras ANALIZE 7.5 y se guardan con `save_untouch_nii`.

## 2.3. Otras funciones para representar la MRI

Existen multitud de programas y funciones en Matlab para representar las MRIs. En este trabajo se han utilizado las funciones de Matlab `imshow`, `image` e `imagesc` para comprobar cómo se representa la matriz de la imagen de resonancia magnética con estas funciones. Además, se han visualizado con un software denominado MRIcro [35].

### 2.3.1 `imshow`, `imagesc`, `image`

La función `imshow` representa una imagen en escala de grises en una figura, optimizando los ejes y las propiedades de la imagen. Es necesario especificar el rango de representación como un vector de dos elementos, siendo el primer elemento el valor más bajo de la matriz de la imagen, y el segundo el valor más alto de la matriz de la imagen. El valor más bajo lo identifica con el color negro, y el más alto con el blanco. Los valores entre este rango se muestran como tonos intermedios de gris, usando la cantidad predeterminada del nivel de gris. Si se especifica una matriz vacía como rango, `imshow` interpreta que el valor bajo es el mínimo de la matriz de la imagen que se quiere representar, y el valor alto con el máximo de esa matriz.

Usar `imshow` para representar la imagen requiere que antes sea cargada por `load_nii`, así se puede acceder a la matriz `img` que contiene el archivo NIFTI. Esta función es válida para matrices de dos dimensiones, por lo que para representar esa matriz es necesario realizar un corte en el eje `z` (véase la figura 14).

```
t1 = load_nii('T1.nii')
imshow(t1.img(:,:,120), [])
```

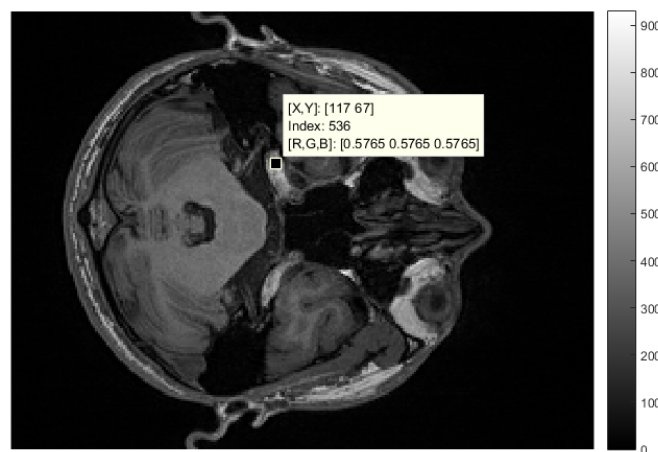


Figura 14. Representación de T1 con `imshow`

La función `imagesc` muestra los datos de una matriz usando la gama completa de colores en el mapa de colores. Cada elemento de la matriz especifica el color para un píxel de la imagen. La imagen resultante es una cuadrícula de `m` por `n` píxeles donde `m` es el número de columnas y `n` es el número de filas de la matriz. Los índices de fila y columna de los elementos determinan los centros de los píxeles correspondientes. Esta función también se usa para matrices de dos dimensiones, por lo que si se quiere representar una matriz de tres dimensiones, como es el caso, se tiene que representar un corte del eje `z`, de la misma forma que con `imshow`,

con la salvedad de que no es necesario indicar el rango en el que se va a representar la imagen (véase la figura 15).

```
t1 = load_nii('T1.nii')
imagesc(t1.img(:,:,120))
```

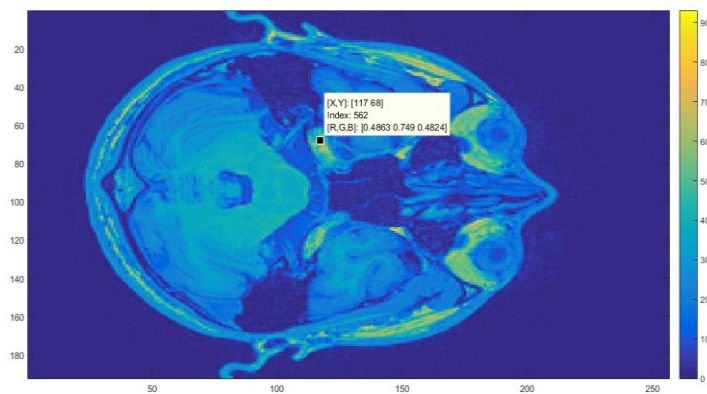


Figura 15. Representación de T1 con `imagesc`

La función `image` es similar a `imagesc`, salvo que ésta muestra los datos en una matriz como una imagen (véase la figura 16).

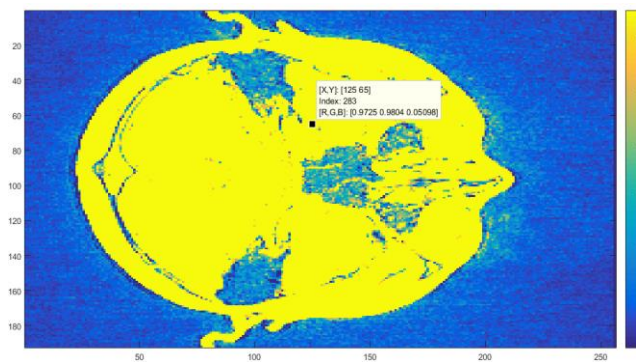


Figura 16. Representación de T1 con `image`

La principal diferencia entre `image` e `imagesc` es que el primer método muestra la matriz de entrada como una imagen y cada valor de la entrada se interpreta directamente como un índice de un color en un mapa recortado de 64 colores. En cambio, `imagesc` aplica un escalado automático con un rango de imagen que abarca todo el mapa de colores completo.

Cabe destacar que, si se selecciona un píxel de la imagen en una determinada coordenada  $(x, y, z)$ , éste se corresponde con el valor de la matriz en las coordenadas  $(y, x, z)$ . Esto quiere decir que los valores de la imagen representada con alguna de estas funciones, se corresponde con los valores de la matriz `img` [36] contenida en el archivo NIFTI. El siguiente ejemplo muestra el acceso a la matriz en las coordenadas  $x$  e  $y$  mostradas en la figura 16.

```
T1.img(65,125,120)
ans = 283
```

Además de esta particularidad de tener invertidos los ejes  $x$  e  $y$ , si se analizan las propiedades de la imagen [37], después de haber sido cargada con alguna de estas tres funciones, se observa que los valores iniciales de los ejes  $x$  e  $y$  no se corresponden con cero sino con 0.5, tal y como se muestra en la figura 17. La dimensión de esa matriz es de 192x256x256; teniendo en cuenta que se invierten los ejes  $x$  e  $y$ , se puede observar que el eje  $x$  empieza en 0.5 y termina en 256.5.

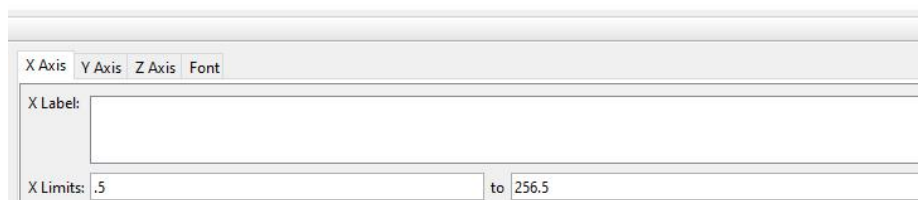
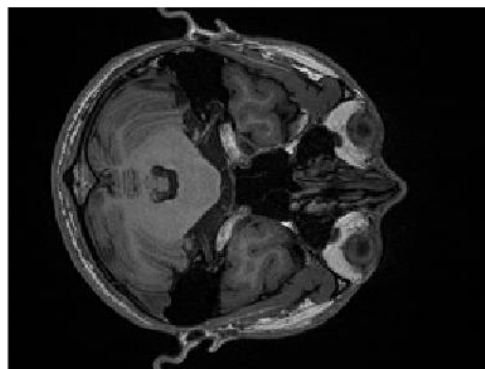


Figura 17. Escalado de ejes en imshow

Esta diferencia se debe a un sistema de coordenadas interno de la imagen. Para acceder a ubicaciones de la imagen, la herramienta Image Processing Toolbox de Matlab usa varios sistemas de coordenadas de imagen diferentes como convenciones para representar imágenes como matrices. El método usado por defecto para expresar ubicaciones en una imagen en Matlab es el de coordenadas espaciales. Se trata de un sistema de coordenadas continuamente variable. Esto le permite considerar que cubre una cuadrícula y se describen en términos de  $x$  e  $y$ , y no de fila y de columna. Dentro de este sistema de coordenadas espaciales existen dos sistemas: coordenadas intrínsecas y coordenadas mundo. El primer sistema, mostrado en la figura 18, se corresponde a índices de píxeles, y el segundo relaciona la imagen con algún otro espacio de coordenadas. Las funciones definidas previamente se basan en el sistema de coordenadas intrínsecas, por defecto. En este sistema de coordenadas  $y$  aumenta hacia abajo y  $x$  aumenta hacia la derecha porque esta orientación es consistente con la forma en que se ven típicamente las imágenes digitales.

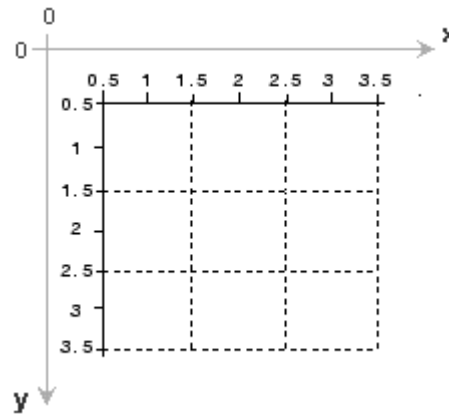


Figura 18. Sistema de coordenadas intrínseca de la imagen

Las coordenadas intrínsecas  $(x, y)$  del punto central de cualquier píxel son idénticas a los índices de columna y fila para ese píxel. Esto significa que el orden de las coordenadas se invierte en las coordenadas intrínsecas relativas a los índices de píxeles, las columnas son las  $x$  y las filas con las  $y$ . En la figura 18 se observa que la esquina superior izquierda de la imagen se encuentra en  $(0.5, 0.5)$  y no en  $(0, 0)$ , y la esquina inferior derecha de la imagen se encuentra en  $(\text{numCol} + 0.5, \text{numFil} + 0.5)$ , donde  $\text{numCol}$  y  $\text{numFil}$  son el número de columnas y filas en la imagen. Por el contrario, el píxel superior izquierdo es  $(1, 1)$  y el píxel inferior derecho es  $(\text{numCol}, \text{numFil})$ . El centro del píxel superior izquierdo es  $(1.0, 1.0)$  y el centro del píxel inferior derecho es  $(\text{numCol}, \text{numFil})$ . De hecho, las coordenadas centrales de cada píxel son valores enteros. El centro del píxel con índices  $(r, c)$  donde  $r$  y  $c$  son enteros por definición, cae en el punto  $x = c, y = r$  en el sistema de coordenadas intrínsecas.

### 2.3.2 MRICron

MRICron es un software gratuito para visualizar imágenes en formato NIFTI multiplataforma que está desarrollado en Windows, Linux y Macintosh OSX [35]. Puede cargar múltiples capas de imágenes en dos dimensiones, generar representaciones de volúmenes y dibujar regiones de interés y estadísticas. También proporciona la función `dcm2nii` para convertir archivos DICOM a NIFTI [38]. Esta herramienta es útil para saber en qué sistema de coordenada se representa la imagen o si está normalizada o no.

MRICron presenta un menú en la barra de herramienta con multitud de funcionalidades. Los números en  $X/Y/Z$  establecen un corte de la imagen, donde  $X$  se refiere a izquierda/derecha [39],  $Y$  se refiere a anterior/posterior y  $Z$  a inferior/superior. El ajuste de estos valores cambiará la visión sagital, coronal y axial que se muestra de la imagen. El siguiente elemento es el zoom. En esta opción las imágenes se escalan de diferentes formas, escalando la vista del cerebro para llenar el panel de manera óptima. Hay opciones como `x1` (100%), `x2` (200%), `x3` (300%) o `fit`, que sitúa la imagen centrada en la pantalla, como se muestra en la figura 19.



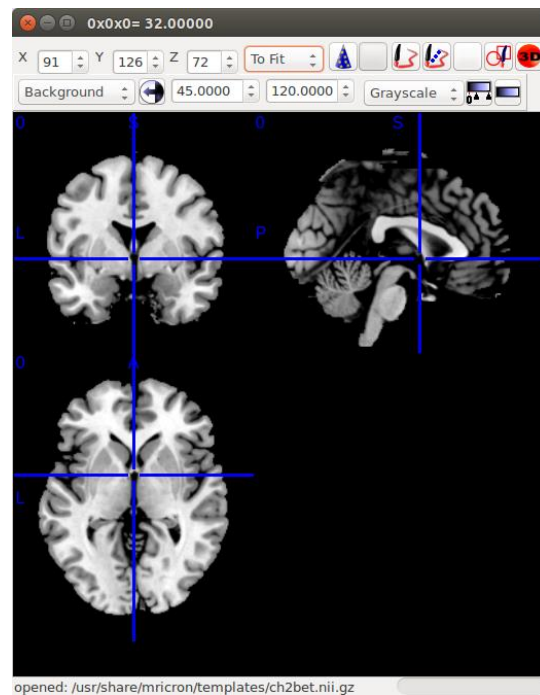


Figura 19. Plantilla de MRIcron con opción de zoom fit

La siguiente serie de botones se refiere a la región de interés cerebral que se muestra en color rojo [40]. Todas estas regiones se deben dibujar con respecto a una misma plantilla, teniendo en cuenta las dimensiones y la orientación de la imagen. El botón “pen tool” puede dibujar, borrar una línea fina o gruesa. “Closed pen tool” dibuja o borra una línea cerrada que puede ser gruesa o fina. “Fill tool” rellena una región de interés y “Circle tool” puede dibujar una elipse o un rectángulo.

MRIcron dispone de una plantilla (template) normalizada al espacio MNI152 [41], construida a partir de la media de las MRI de 152 sujetos. Si se abren las propiedades de esta imagen se puede observar muchos de los parámetros que se describieron en el apartado del formato NIFTI, como por ejemplo el tipo de cabecera, las dimensiones de la imagen, etc. A continuación, se describen las propiedades de la plantilla que dispone MRIcron.

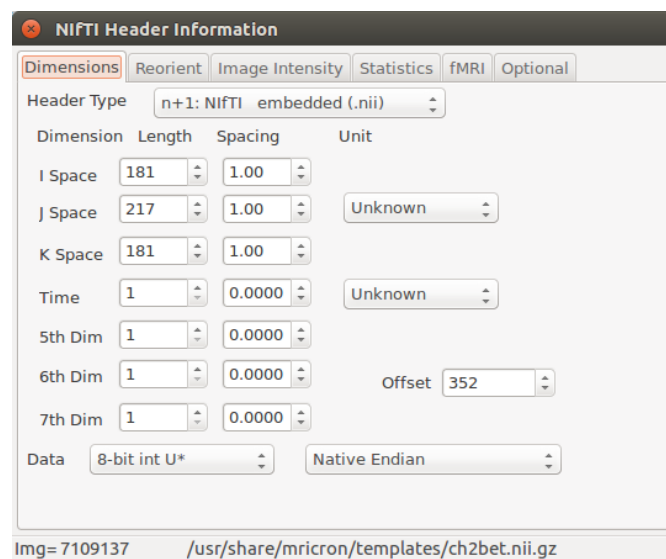


Figura 20. Información de cabecera NIFTI, sección Dimensions



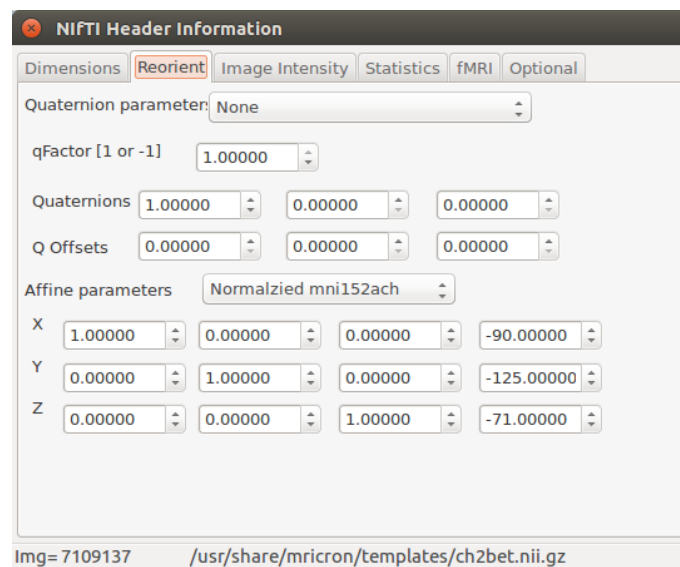


Figura 21. Información de cabecera NIFTI, sección Reorient

En la sección “Dimensions” (véase la figura 20) se observa que la extensión del archivo es `.nii`, y se corresponde con el valor de `magic` del toolbox de NIFTI ( $n+1$ ). También se muestran las dimensiones de la imagen (181x217x181) y el tamaño del vóxel (1 mm). En la sección “Reorient” (figura 21) se encuentra información acerca de la matriz afín, y del sistema de coordenadas utilizado para normalizar la imagen (Normalized mni152ach).

Existe una opción “Overlay” que permite añadir una segunda imagen superpuesta. Esto sirve para comprobar si una imagen está normalizada. En caso afirmativo, la imagen quedará representada encima de la otra imagen normalizada, haciendo coincidir las partes del cerebro. A continuación, se muestran dos figuras: en la primera (figura 22) se ha representado una imagen normalizada al espacio MNI, y se ha añadido otra imagen que ha sido segmentada. Se observa como esta imagen queda perfectamente superpuesta, es decir, está normalizada; en la figura 23 se ha representado la misma imagen normalizada y se ha añadido una imagen T1 de un sujeto. Se puede observar que esta imagen queda desplazada, por lo que no está normalizada. Al acceder la información de la cabecera NIFTI se comprueba que, efectivamente, el sistema de coordenadas de esta imagen es el de la máquina de la que se extrajo la MRI.

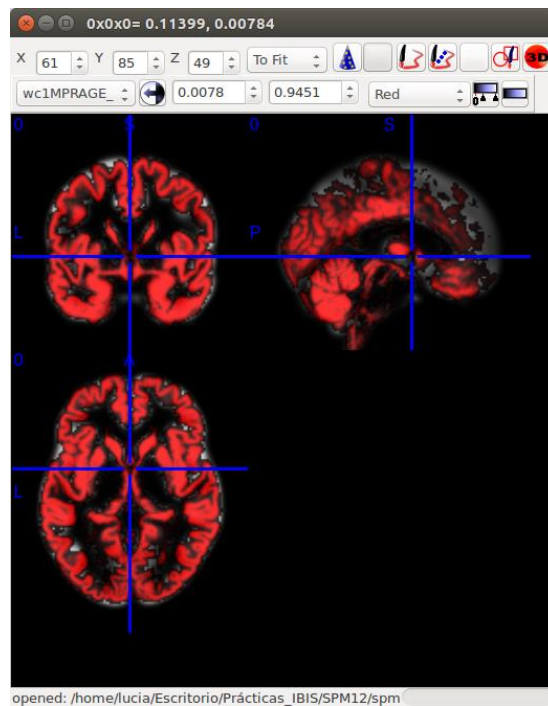


Figura 22. Imagen normalizada sobre plantilla normalizada

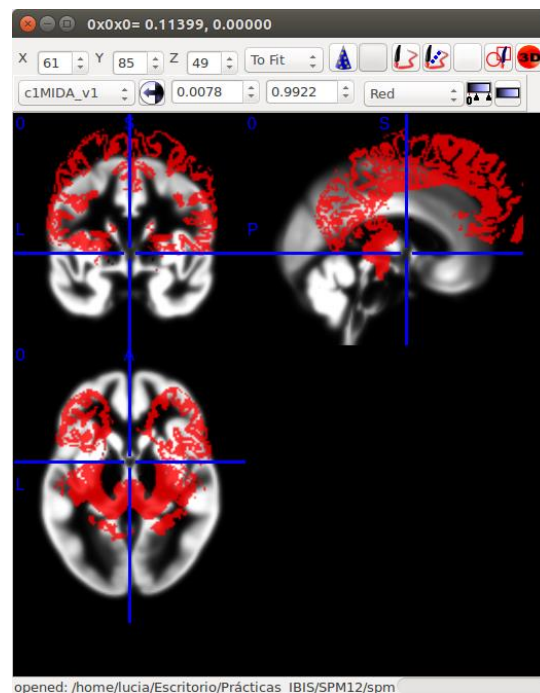


Figura 23. Imagen no normalizada sobre plantilla normalizada

Otra opción es abrir dos ventanas de MRIcron: en una ventana representar la imagen que está normalizada, y en la otra representar la imagen segmentada. Al situar el cursor en una posición determinada en una imagen, debe coincidir con el de la otra para que esté normalizada. La figura 24 muestra un ejemplo de imagen normalizada frente a una que no lo está. Se observa que el cursor en la imagen no normalizada se encuentra en la comisura anterior, mientras que en la imagen normalizada se encuentra en un borde de la materia gris.

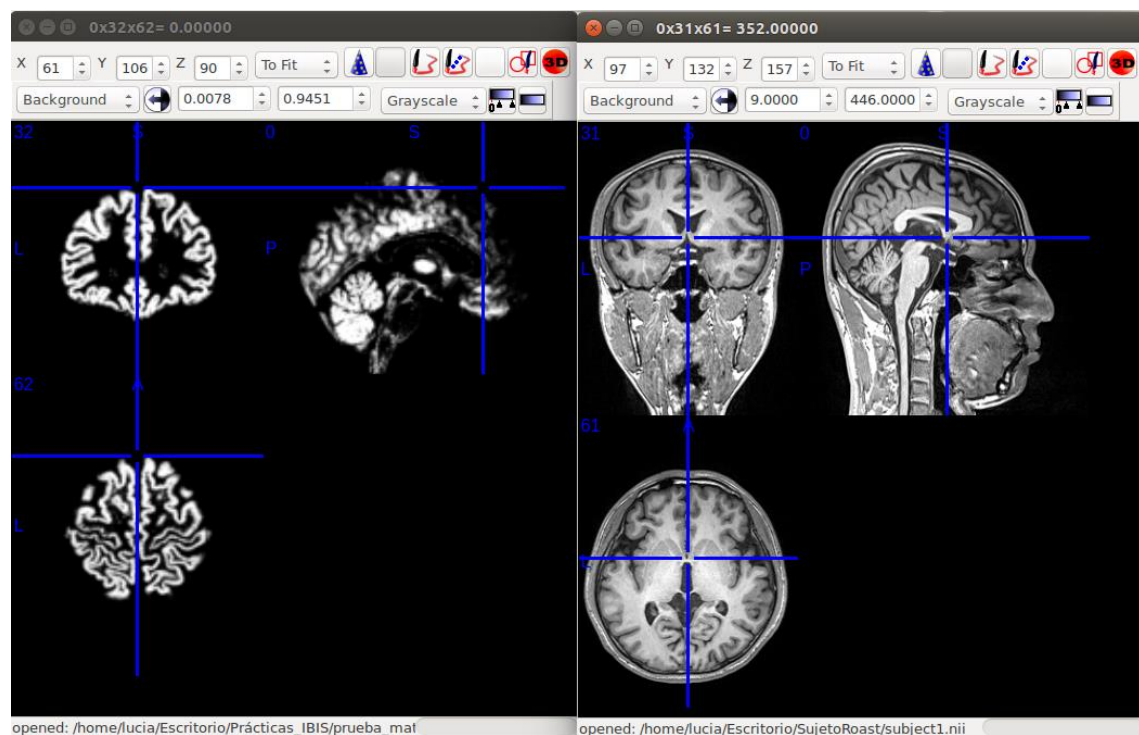


Figura 24. Plantilla normalizada a la izquierda. MRI no normalizada a la derecha

El archivo NIFTI almacena transformaciones espaciales para que MRICron pueda determinar la orientación de la imagen de forma intuitiva, a diferencia de otras herramientas que ignoran estas transformaciones. La figura 25 resume tres orientaciones posibles.

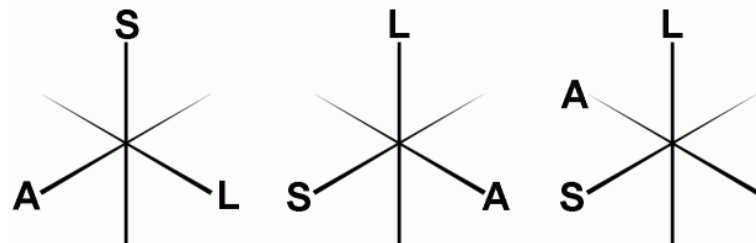


Figura 25. Ejemplos de orientaciones posibles en MRICron

Una opción de MRICron que a veces puede ser útil, es la posibilidad de invertir los ejes x e y de la imagen. Esto es posible con la opción **Flip L/R** del menú de la barra de herramientas **View**. La figura 26 muestra una imagen cerebral con los ejes invertidos.

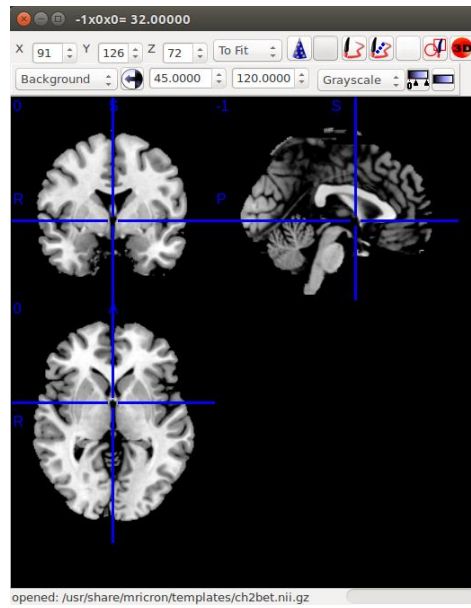


Figura 26. Ejes x e y invertidos.

### 2.3.3 MRIs utilizadas para el estudio

Se han elegido dos imágenes T1 y T2 de la web PPMI (Parkinson's Progression Markers Initiative [42]) cuyas dimensiones son 170x240x240 y 170x256x256, y que se muestran en las figuras 27 y 28, respectivamente.

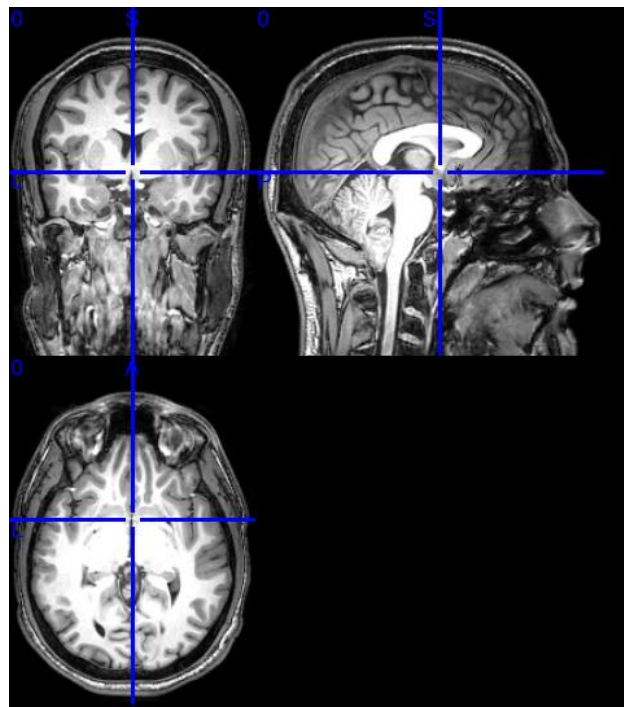


Figura 27. T1.nii descargada de la web PPMI y representada con MRICron

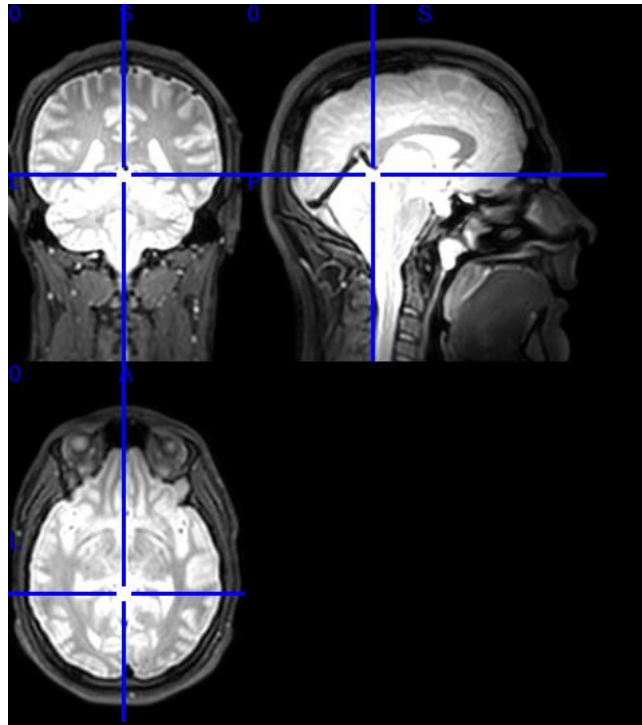


Figura 28. T2.nii descargada de la web PPMI y representada con MRIcron

## 2.4. Metodología

En este subapartado se explicarán los distintos pasos seguidos para la elaboración de un flujo de trabajo de utilidad para modelos computacionales del cerebro. Estos son la segmentación de la imagen de resonancia magnética, el mallado superficial [43] y volumétrico de los tejidos, y la posterior colocación de electrodos sobre la piel. Para cada uno de estos pasos se estudiará distintas herramientas, de las cuales se explicará en qué consisten, cómo proceder para su uso, así como algunas funcionalidades importantes.

### 2.4.1 Segmentación

La segmentación [44] es el proceso que consiste en simplificar una imagen dividiéndola en varias partes. Este proceso se realiza asignando una etiqueta a cada píxel de la imagen. El resultado de la segmentación de una imagen es la misma imagen dividida en regiones debido a que los píxeles de cada región tienen unas características similares como el color, la intensidad o textura. De entre las diferentes herramientas software que existen para segmentar las imágenes biomédicas, en este trabajo se han considerado SPM y SIMNIBS, debido a que SPM segmenta hasta el cuello y está implementado en Matlab, y con SIMNIBS se obtiene muy buena segmentación de la materia blanca y gris.

#### 2.4.1.1 SPM

SPM (Statistical Parametric Mapping) [45] es un software gratuito de funciones y subrutinas de Matlab con algunas rutinas C compiladas externamente., distribuido bajo los términos de la Licencia Pública General (GNU) publicada por la Fundación de Software Libre. Es una técnica de construcción y evaluación de procesos estadísticos espacialmente utilizados para probar hipótesis acerca de los datos de imágenes funcionales. Analiza secuencias de datos de imágenes cerebrales, como fMRI (imagen de resonancia magnética funcional), PET (Positron Emission Tomography), SPECT (Single Photon Emission Computed Tomography), EEG (Electroencefalografía) y MEG [46] (Magnetoencefalografía).

Existen varias versiones de este programa: SPM99, SPM2, SPM5, SPM8 y SPM12 [45]. En este trabajo se usará

la versión más reciente, SPM12 [47], por sus importantes actualizaciones en los algoritmos, plantillas [48] y modelos utilizados. SPM12 usa el formato NIFTI y, a diferencia de versiones anteriores, el formato de ficheros por defecto es .nii, en lugar de .hdr/.img.

Con SPM12 se pueden segmentar varias imágenes simultáneamente. Se puede segmentar una imagen T1 y obtener sus tejidos en formato .nii, u optar por segmentar una imagen T1 junto con una T2. Con esta última opción se obtienen mejores resultados en el cráneo [49]. El proceso que hay que seguir en los dos casos es distinto. Para la primera opción basta con reorientar T1 para que el origen de coordenadas se sitúe en la comisura anterior, sin embargo, para la segunda opción hay que reorientar las dos imágenes [50], hacer un correregistro de ambas y alinearlas [51]. En el siguiente apartado se explica el procedimiento de segmentar dos imágenes T1 y T2, ya que este segundo caso engloba al primero.

### Segmentación de T1 y T2 con SPM12

Para que la segmentación de T1 y T2 sea posible, las imágenes deben tener el mismo centro de coordenadas situado en la comisura anterior, tener la misma dimensión y estar alineadas, es decir, que las imágenes coincidan perfectamente en sus centros.

Al iniciarse el programa (figura 29), se elige la opción fMRI y a continuación aparece un menú en el que se deben seleccionar, las opciones “Coregister and reslice” en la parte de preprocesado espacial y “Display” en la parte de fMRI para SPM [52].

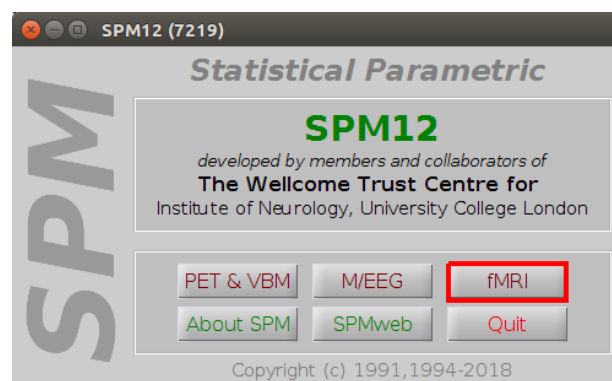


Figura 29. Pantalla de inicio SPM12

La función “Coregister” (figura 30) usa la información mutua para hacer coincidir la imagen T1 con la T2. Esta función solo permite translación y rotación de la imagen. En la opción de “Coregister and reslice” se debe seleccionar la imagen con mejor resolución como Reference (T1 o T2) y la otra como Source (si en Reference fue seleccionada T1, en Source se selecciona T2). Al terminar este proceso se obtiene una imagen llamada rT2 (en el caso de haber puesto a T2 como imagen fuente, con la misma dimensión que tenga la imagen de referencia y alineada a ella [53]).

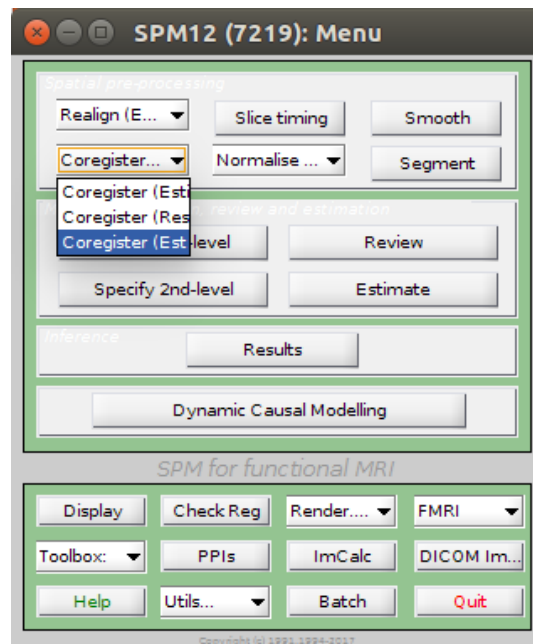


Figura 30. Corregistro y alineación de imágenes

Ahora se puede proceder a reorientar las imágenes, imponiendo como origen de coordenadas la comisura anterior. Para ello se selecciona la opción “Display” del menú principal (véase la figura 31) y se visualiza, en primer lugar, T1. Se selecciona la comisura anterior del cerebro y se guarda la nueva orientación marcada en la casilla “vx” (ver figura 30), pinchando en “Set origin” y “Reorient”. Esto guardará un archivo con extensión .mat con la nueva orientación de la imagen. A continuación, se realiza el mismo procedimiento para rT2 (elegir rT2 y no T2, ya que rT2 está corregistrada y alineada a T1).

Es muy importante que el origen en T1 coincida exactamente con el origen en rT2 para que la segmentación se haga correctamente. Al seleccionar con el cursor en la comisura anterior, las coordenadas tienen más decimales de los que se muestran en la casilla “vx”, por lo que se recomienda escribir manualmente las coordenadas de la comisura anterior en dicha casilla, y luego copiar y pegar estas coordenadas en la casilla “vx” de la imagen rT2. De esta forma, ambas coordenadas coinciden perfectamente.

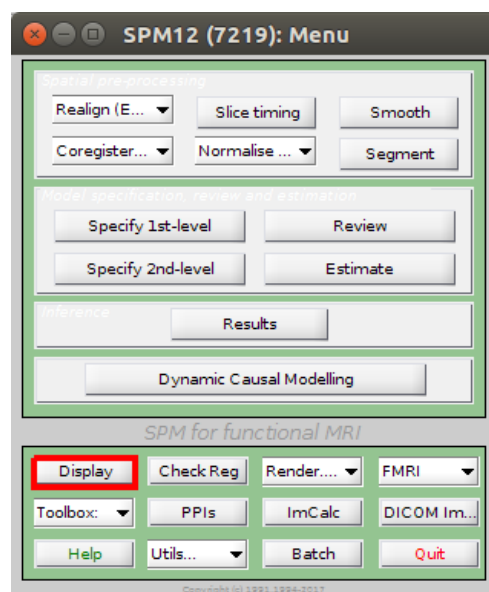


Figura 31. Display para visualizar las imágenes



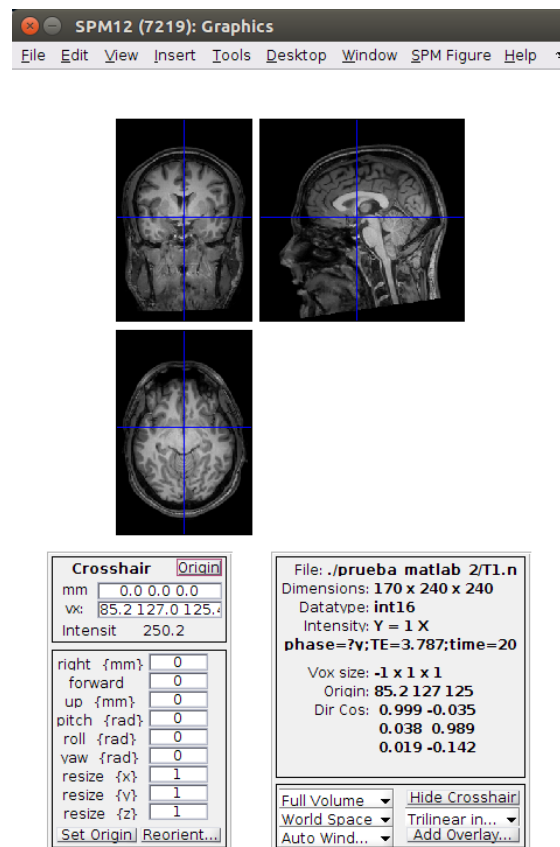


Figura 32. Opciones del Display

Las opciones más importantes que presenta el “Display” son (véase la figura 32):

- Crosshair indica la posición de la línea azul en la imagen en milímetros y en vóxeles.
- Right, forward y up mueve en tres direcciones la imagen.
- Pitch, roll y yaw rota la imagen alrededor de los ejes x y z, respectivamente.
- Reorient: en este botón se guardan los cambios que se le hayan hecho a la imagen. En el lado derecho se puede observar alguna información sobre la imagen, como el tamaño del vóxel, el origen.

Para segmentar las imágenes se selecciona “Segment” del menú principal (véase la figura 33), que abrirá un “Batch Editor” (figura 34) donde hay que introducir en “Volumes” las dos imágenes que serán segmentadas. Este proceso tardará pocos minutos y al finalizar, se tendrán las segmentaciones de las imágenes en la misma carpeta donde se tenga almacenada la T1.



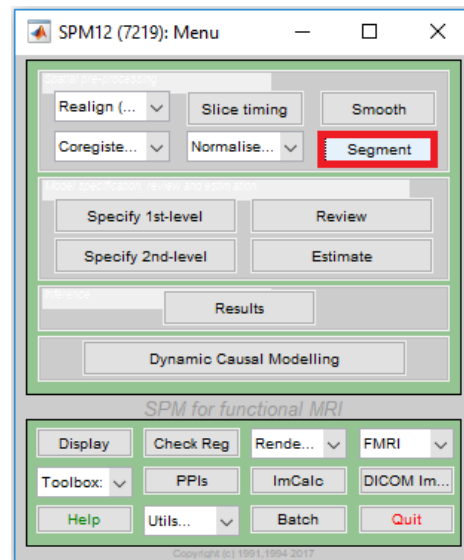


Figura 33. Opción Segment del menú principal

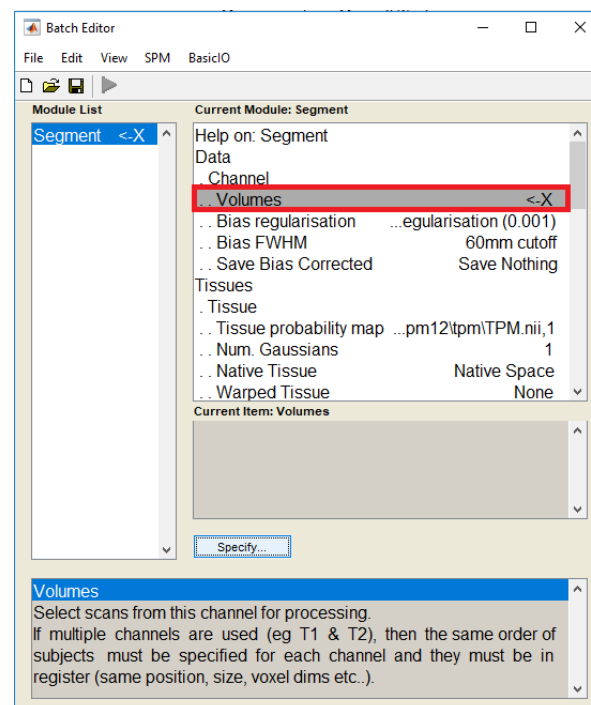


Figura 34. Batch Editor para Segment

SPM12 dispone de la plantilla TPM.nii para segmentar las imágenes, que contienen todos los tejidos que van a ser segmentados. Al finalizar la segmentación, se obtienen seis segmentaciones, para cada imagen, nombradas con los prefijos c1, c2, c3, c4, c5 y c6. c1 se corresponde con la materia gris, c2 es la segmentación de la materia blanca, c3 es el fluido cerebroespinal (líquido cefalorraquídeo, CSF), c4 es el cráneo, c5 se corresponde con el tejido de la piel y c6 es el aire. Los tejidos son archivos con extensión .nii y se pueden representar en SPM12 con “Check reg”.

SPM12 dispone de una opción llamada “smoothing” o suavizado [54]. El suavizado espacial generalmente se realiza como parte del preprocesamiento de imágenes cerebrales [55]. En SPM el suavizado espacial se realiza

con un filtro Gaussiano fijo en el espacio donde el usuario debe especificar el ancho del núcleo en mm. El suavizado reduce la resolución de la imagen resolución y aumenta la relación señal a ruido de cada vóxel. Cuanto menos se suavice, menos resultados significativos se obtienen.

Por último se pueden normalizar [56] las imágenes al espacio MNI seleccionando la opción “Normalize”. De esta forma, la imagen se normaliza usando una plantilla de SPM [57]. El problema es que la plantilla no es universalmente representativa y los resultados no son muy precisos. Otra forma más precisa para pasar al espacio normalizado es crear una plantilla propia con las segmentaciones realizadas anteriormente. Esto es posible con DARTEL [58] que aplica varios avances para abordar esas limitaciones.

### Segmentación con ROAST

También se ha realizado el estudio del toolbox de ROAST (Realistic Volumetric Approach to Simulate Transcranial) [59]. Es una herramienta desarrollada en Matlab con funciones de SPM desarrollada por el laboratorio de Lucas C. Parra para simular la estimulación eléctrica transcraneal de forma automática.

ROAST realiza la segmentación con una función llamada `start_seg` cuyos argumentos de entrada son: T1, T2 (opcional), `template` (opcional), normalización (opcional). Dado que `start_seg` usa funciones de SPM, es necesario hacer todo el preprocesado explicado anteriormente para las imágenes T1 y T2 (corregistro y reslice, y reorientar [60]). En el caso de realizar la segmentación con SPM12 directamente, se recomendaba ese procedimiento para obtener una buena segmentación de las MRIs. Sin embargo, con ROAST la segmentación de T1 y T2 será imposible si ese procedimiento no se realiza correctamente.

El primer argumento de entrada de la función `start_seg`, T1, debe estar en formato `.nii`. En el caso de no introducir ninguna imagen, es decir, el número de argumentos de entrada es cero, se abrirá un cuadro de diálogo para que el usuario seleccione la MRI que desee. El segundo argumento de entrada, T2, es opcional. Puede realizarse una segmentación únicamente con la imagen T1. La tercera entrada es el `template` o plantilla que, en caso de no indicar ninguna, ROAST tiene una por defecto. Por último, se pueden normalizar los tejidos segmentados si el cuarto argumento es `true`. En caso de no indicar nada, no se normalizará.

ROAST permite la segmentación de T1 y T2 combinando ambas [61], de esta forma se obtienen mejores resultados de segmentación. Además, esta herramienta contiene un `template` propio llamado `eTPM` con el cual se obtienen resultados muy mejorados, sobre todo en el fluido cerebroespinal y en el cráneo. Por último, también es capaz de crear las máscaras de 0 y 255 de cada uno de los tejidos segmentados mediante la función `mysegment`.

#### 2.4.1.2 SIMNIBS

SIMNIBS es un paquete de software gratuito, desarrollado en linux, para la simulación de estimulación cerebral no invasiva. Permite cálculos realistas del campo eléctrico inducido por estimulación magnética transcraneal (TMS [62]) y estimulación de corriente directa transcraneal (tDCS). SIMNIBS usa FreeSurfer para segmentar el cerebro y FSL BET/BET surf [63] [64] para segmentar el resto de la cabeza. FreeSurfer [65] necesita una buena separación entre el cerebro y la señal del hueso para que haga una buena segmentación. FreeSurfer es un paquete de software de código abierto para el análisis y la visualización de datos de neuroimágenes estructurales y funcionales a partir de estudios transversales o longitudinales. FSL es un software que contiene análisis de imágenes y herramientas estadísticas para datos de imágenes cerebrales de MRI funcionales, estructurales y de difusión. FSL está disponible como código fuente para Apple y Pc (Linux) gratuito para uso no comercial.

SIMNIBS crea modelos de la cabeza a partir de MRIs en formato NIFTI, mediante el comando `mri2mesh`. Esta es la herramienta principal para reconstruir un modelo de cabeza a partir de T1 y T2. Funciona con una sola imagen, pero dará mejores resultados de cráneo si se dispone de una T2. El comando a ejecutar es el siguiente: `mri2mesh --all almi5 org/almi5_T1fs.nii.gz org/almi5_T2.nii.gz`. [66] Con el comando `--all` se ejecutan todos los pasos de reconstrucción incluyendo el mallado de volumen. `almi5` es el identificador del sujeto y se crearán las carpetas `fs_almi5` (contiene los resultados de FreeSurfer), `m2m_almi5` (contiene todos los ficheros que fueron necesarios para la creación del volumen, como las

segmentaciones en formato NIFTI) y el archivo `almi5.msh`, que es la malla creada [67]. Este proceso puede tardar varias horas dependiendo de la memoria RAM de la que disponga el equipo. Para un equipo de 8 GB de RAM y Core i7 tarda unas once horas.

## 2.4.2 Malla superficial

Una malla triangular es un tipo de malla que comprende un conjunto de triángulos, generalmente en tres dimensiones, que están conectados por sus bordes o por sus esquinas comunes. Una malla de superficie triangular [68] es la base de cualquier representación de superficie incluyendo varios métodos numéricos en ingeniería eléctrica y biomédica, gráficos de computadora, etc. Se considera un cuadrado de tamaño  $a \times b$  en un plano de dos dimensiones  $xy$ . El objetivo es rellenar ese cuadrado con triángulos definiendo los nodos o vértices de cada triángulo en una matriz. Esta matriz, de dimensión  $\text{nodos} \times 3$ , contiene por cada fila tres coordenadas cartesianas correspondientes a cada nodo. El número de fila en la matriz es el número del vértice. Además, se necesita otra matriz en la que se definan las caras del triángulo. Esta matriz, de dimensión  $\text{triángulos} \times 3$ , contiene tres números enteros de los vértices de los triángulos.

Para hacer las mallas superficiales se ha empleado un toolbox desarrollado en Matlab llamado `iso2mesh`. `iso2mesh` es una caja de herramientas de generación y procesamiento de malla libre basada en matlab / octave. Puede crear una malla tridimensional tetraédrica de elementos finitos (FE) a partir de superficies, imágenes volumétricas en 3D binarias y en escala de grises. `Iso2mesh` es un software de código abierto desarrollado por Qianqian Fang. Es profesor asistente en el Departamento de Bioingeniería de la Universidad Northeastern [69]. En la figura 35 se resume el flujo que sigue `iso2mesh` [70] para la creación de mallas volumétricas a partir de una MRI.

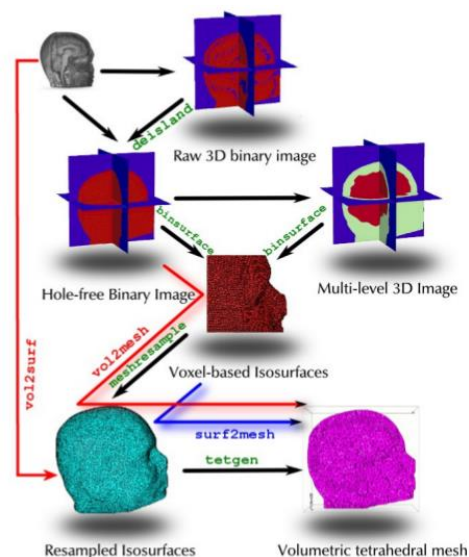


Figura 35. Esquema de ISO2MESH para creación de mallas volumétricas

En este apartado se explicará el proceso hasta la creación de la malla superficial. Para la realización de la malla existen dos caminos representados en la figura anterior: un camino es usando `binsurface`, y el otro es usando `vol2surf`.

### Binsurface

Para la primera opción, se usa la función `binsurface` para generar las mallas superficiales a partir de una imagen binaria de 0 y 1. Primero se carga la malla en formato `.nii` con la función `load_untouch_nii` del toolbox de NIFTI y se extrae la imagen `.img`. Luego se cierran los huecos de la imagen rellenándola con 1,

usando la función `fillholes3D` y se eliminan los píxeles de la imagen que están aislados, con la función `deisland3D`. Ambas funciones pertenecen al toolbox de `iso2mesh` y tienen un segundo parámetro de entrada definido como el máximo tamaño del hueco para cerrar las imágenes (con un valor de 10 en este trabajo). El problema que presenta esta opción es que se queda con la superficie externa y cierra los agujeros. En algunos tejidos funciona muy bien, pero en otros como el cráneo, cierra los huecos de los ojos y no se obtienen mallas muy realistas. Ahora se puede proceder a la creación de los nodos y las caras de los triángulos (fáces) con la función `binsurface`, a la cual se le pasan como argumentos de entradas la matriz de la imagen binaria que ha sido modificada con las funciones anteriores, y un segundo parámetro numérico, 3, que indica que las caras son triangulares. Se comprueba la malla superficial con la función `meshcheckrepair` y se remuestra la malla con la función `meshresample` utilizando el método CGAL con la utilidad de simplificación.

## Vol2surf

La segunda opción es generar directamente la superficie de triángulos a partir de las máscaras de todos los tejidos en formato `.nii`, con la función `vol2surf` de la herramienta `iso2mesh`. Esta función convierte una imagen en 3 dimensiones binaria a una superficie y se le debe indicar el valor del tejido que se desea reconstruir. `vol2surf` dispone de algunos parámetros para la extracción del mallado de superficie, como son: `simplify` y `cgalsurf`. En el primer caso se crea una malla triangular en resolución de vóxel y luego se aplica un proceso de simplificación basada en la densidad de la malla superficial especificada por el usuario (puede incluir múltiples superficies desconectadas). Para realizar este procedimiento `vol2surf` llama a `binsurface` para generar la superficie basada en vóxel y a `meshresample/meshcheckrepair` para crear la malla. En el segundo caso se crea la superficie triangular directamente de la imagen volumétrica de entrada, usando una versión modificada del programa de extracción de superficie, CGAL.

### 2.4.3 Malla volumétrica

Las mallas volumétricas son una representación poligonal del volumen interior de un objeto. Comprende un conjunto de tetraedros conectados por sus cuatro bordes o esquinas comunes. Una malla volumétrica [71] queda determinada por sus nodos, caras y elementos. Los nodos y las caras están definidos en el apartado de malla superficial. La matriz de elementos, de dimensión `elemx5`, contiene las filas de los vértices que forman cada tetraedro. La última columna se corresponde con la etiqueta del tejido correspondiente.

La figura 35 muestra el esquema seguido con la herramienta `iso2mesh` para la creación de mallas volumétricas a partir de una MRI. Para la creación de mallas volumétricas es necesario tener previamente la malla superficial correspondiente. Del esquema se puede observar el método `surf2mesh` para formar la malla volumétrica a partir de la superficial, el cual se explicará en el siguiente subapartado.

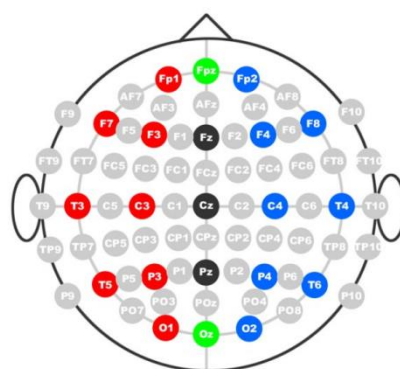
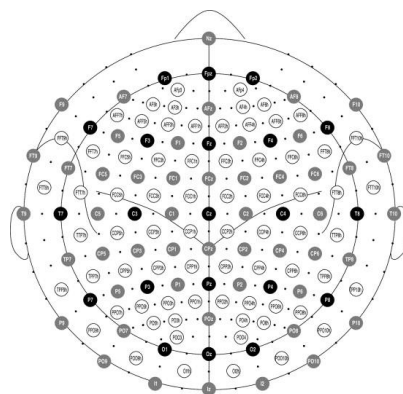
## Surf2mesh

`surf2mesh` crea una malla volumétrica desde una superficie. Se trata de una función perteneciente a la herramienta `iso2mesh`. Para ello, se necesitan las matrices de nodos y caras que forman los triángulos de la superficie de cada uno de los tejidos. En primer lugar se deben fusionar las mallas triangulares de los tejidos con la función `mergesurf`. `surf2mesh` llama a un generador de malla 3D llamado `tetgen`. Éste crea mallas tetraédricas a partir de superficies previamente cerradas, utilizando puntos interiores a esas superficies.

### 2.4.4 Colocación de electrodos

Después de la malla superficial, se procede a la colocación de electrodos sobre la misma. Para ello se han estudiado dos herramientas distintas que calculan las posiciones de los electrodos en la cabeza del sujeto de acuerdo a las coordenadas de tres sistemas internacionales estándar de colocación de electrodos. Los nombres de las herramientas estudiadas son `Meth` (de `iso2mesh`) y `Mesh2EEG`. Y los sistemas son el 10/05 [72], 10/10, 10/20 [73], [74]. Una vez calculada la posición de los electrodos, se procedió a la reconstrucción de los mismos con una función de `iso2mesh`.





Match es una herramienta dentro de i200match

match es una herramienta dentro de Iso2mesh que sirve para ubicar, sobre la malla superficial de la piel, unas posiciones determinadas por una matriz. Esta herramienta consta de dos ejemplos, uno en forma de código y otro en forma de interfaz gráfica. En primer lugar, se cargan los datos correspondientes a los “nodos”, “faces” y la matriz de las coordenadas correspondientes a las posiciones que se quieren marcar sobre la malla. Se debe realizar un mapeo de estas coordenadas a las coordenadas correspondientes del espacio donde se encuentra la cabeza del sujeto mallada. Para ello, con ayuda de la interfaz gráfica se seleccionan cuatro puntos de referencia sobre la cabeza del sujeto, que son Nasion, Cz, T3 y T4. Y se transforma de un espacio de coordenadas con una transformada afin que mapea los nodos de un espacio a otro utilizando la solución de mínimos cuadrados. Con esto se consigue una nueva matriz de coordenadas en el espacio de la cabeza, pero los puntos que forman cada una de esas coordenadas no están pegados a la superficie. Por ello, se procede a una primera optimización para registrar esa nube de puntos en la superficie triangular, con lo que se consigue que los puntos estén más cerca pero aun no pegados a la superficie. Por último, se hace una segunda optimización haciendo la proyección normal desde esos puntos hacia la superficie.

sistemas 10/20, 10/10 y 10/05. Para ello se crearon cuatro archivos .mat, tres para las matrices de cada uno de los sistemas y un cuarto que contenían los “nodos” y “faces” del sujeto. En la figura 40 se presenta el mallado triangular de la piel con el sistema 10/20 cargado. Se muestran tres tipos de puntos de distintos colores. Los puntos rojos se corresponden con el mapeo inicial, los asteriscos verdes son la optimización de esos puntos y los asteriscos celestes son las posiciones del sistema 10/20 en la superficie de la cabeza del sujeto.

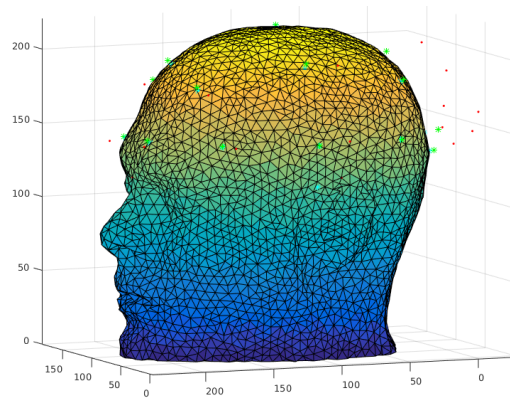


Figura 40. Ubicación de electrodos con la herramienta Metch. Mapeo

El inconveniente que presenta este programa es la alta precisión que se necesita para colocar los cuatro puntos de referencia y conseguir que los puntos finales queden en sus respectivas posiciones sobre la malla. En las figuras 41 y 42 se muestra la malla superficial de la piel del sujeto solo con los asteriscos celestes, representados esta vez con círculos celestes para su mejor visualización. Se puede observar que las posiciones calculadas, en algunos casos, no se corresponden muy bien con las posiciones del sistema 10/20.

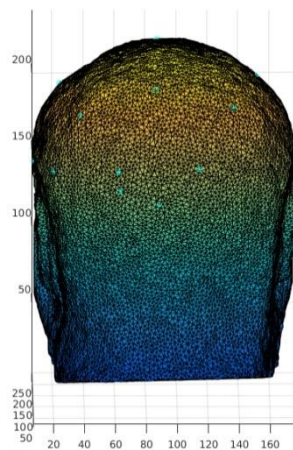


Figura 41. Ubicación de electrodos con la herramienta Metch. Parte trasera de la cabeza

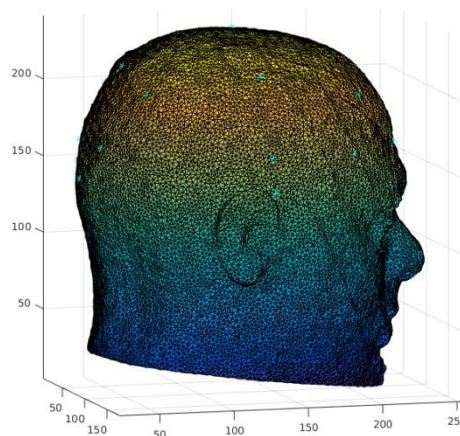


Figura 42. Ubicación de electrodos con la herramienta Metch. Parte lateral de la cabeza



### Herramienta Mesh2EEG para colocación de electrodos

Mesh2EEG es una herramienta que computa las posiciones de los electrodos EEG dadas las caras (faces) y nodos de una malla, y las posiciones de Nasion, Inion, T3 y T4. Primero, con esos cuatro puntos, calcula donde está Cz basada en la posición de los cuatro puntos anteriores. Luego optimiza el valor de Cz en la superficie. Una vez obtenido el punto Cz optimizado, va trazando arcos sobre la cabeza basados en los cuatro puntos de referencia y Cz. Por último, utiliza estos arcos para calcular todo el resto de puntos de las posiciones de EEG. Mesh2EEG tiene la opción de seleccionar el sistema de coordenadas que se quiera usar, el sistema 10/05, 10/10 o 10/20, además de una opción para usar el sistema de coordenadas NIRS correspondiente al 10/10.

También se ha probado esta herramienta con el mismo sujeto de estudio para colocar los electrodos en las posiciones del sistema 10/20, como se observa en las figuras 43 y 44. Cabe destacar que este método es más preciso que el anterior.

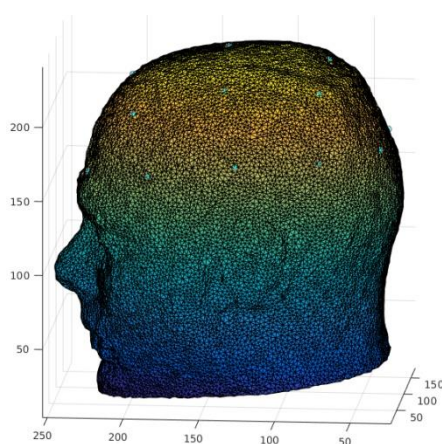


Figura 43. Ubicación de electrodos con la herramienta Mesh2EEG. Parte lateral de la cabeza

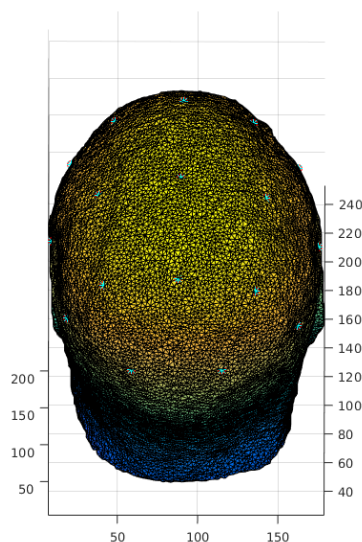


Figura 44. Ubicación de electrodos con la herramienta Mesh2EEG. Parte trasera de la cabeza

El inconveniente que tiene es que los cuatro puntos de referencia hay que introducirlos manualmente. Primero se representa la malla de superficie triangular y sobre ella se marcan los cuatro puntos que deberán igualarse a los valores Nz, Iz, T3 y T4, dentro del código.



### Colocación de electrodos con ROAST

ROAST mapea los marcadores Nz, Iz, T3, T4, parte del cuello frontal y parte trasera del cuello en el archivo TPM.nii. Dado que esos marcadores están definidos sobre el template TPM, ya tienen unas coordenadas fijadas. Luego mapea estas coordenadas al espacio de la MRI. A continuación convierte la imagen a orientación RAS, detecta los puntos que pertenecen a la superficie de la MRI y calcula el lugar de los electrodos a partir de la MRI, de los puntos de la superficie de la MRI y de los marcadores predefinidos. ROAST [59] contiene dos archivos .mat con los sistemas de electrodos EEG 10/05, 10/10, 10/20 [78].

Se pueden elegir los electrodos que se quieren colocar, el tipo de electrodo, que puede ser en forma de cilindro, de rectángulo o incluso anular [79]. Además, si el electrodo seleccionado es el de rectángulo, se puede definir la orientación del mismo. También se puede elegir el tamaño. Es cierto que tiene muchas opciones, pero también tiene un inconveniente, y es que realiza la colocación de electrodos en la MRI del sujeto, y luego lo malla todo. Esta solución es menos eficiente que colocar los electrodos tras realizar el mallado de la piel de la cabeza. En las figuras 45 y 46 se observa que los electrodos son colocados sobre la MRI del sujeto y su posterior mallado del conjunto.

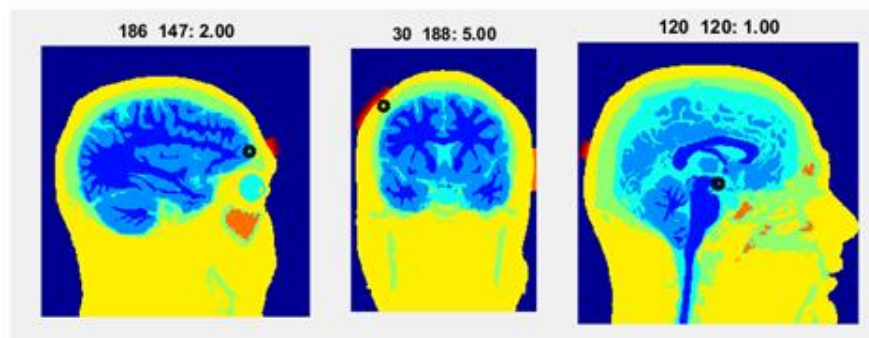


Figura 45. Electrodos colocados con ROAST sobre la MRI



Figura 46. Electrodos y fluido cerebroespinal mallados con ROAST



## 3 RESULTADOS

---

### 3.1 Introducción

En este capítulo se describen los resultados derivados de la investigación realizada en este trabajo. En primer lugar se muestran y explican los resultados obtenidos de las segmentaciones realizadas con SPM12, con ROAST [80] usando distintas plantillas y distintas versiones de SPM, y con SIMNIBS [81]. En segundo lugar se muestran los resultados de las mallas superficiales usando `iso2mesh` con diferentes métodos para las segmentaciones realizadas anteriormente. A continuación se muestran los resultados de las mallas volumétricas.

Por otro lado, se aclara el método utilizado para poder colocar los electrodos sobre la superficie mallada de la piel del sujeto, incluyendo mejoras y la reconstrucción de los electrodos en forma de cilindro. Además se explica paso a paso el procedimiento a seguir para la elaboración del modelo computacional [82] desde las MRIs.

Por último, se aplican todos los conocimientos aprendidos para resolver el sistema de posicionamiento del software Brainsight, internacionalmente utilizado para estudios relacionados con TMS [83].

### 3.2 Resultados de la segmentación realizada con SPM12

Se ha realizado la segmentación de las imágenes de resonancia magnética de estudio con SPM12 ya que SPM está desarrollado en Matlab. Al segmentar con SPM12 se extrae la materia gris, la materia blanca, el fluido cerebroespinal, el cráneo y la piel. Los tejidos segmentados serán renombrados añadiéndole al nombre de la MRI un prefijo `c1`, `c2`, `c4`, `c3`, `c5`, respectivamente. El formato es NIFTI [84] y las coordenadas son las mismas que tiene la MRI [85] que se ha segmentado [86]. A continuación se muestran las segmentaciones realizadas con la T1 y T2.

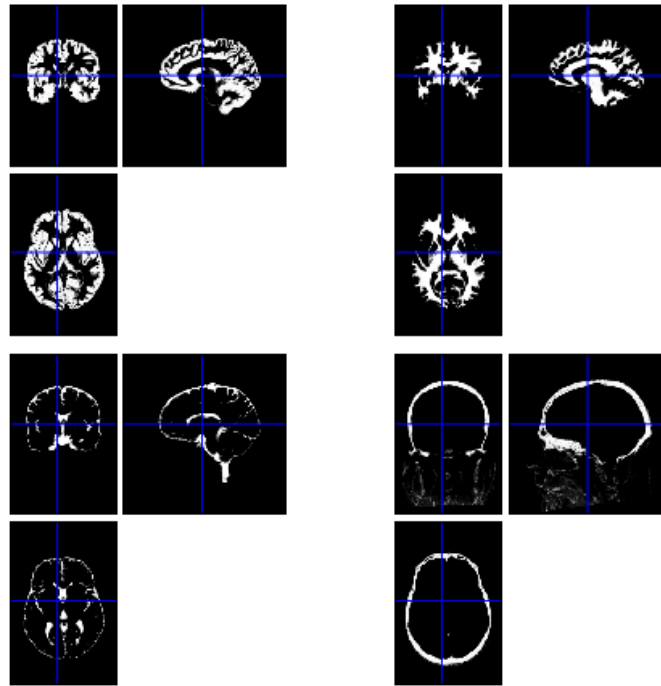


Figura 46. Resultados de la segmentación de T1

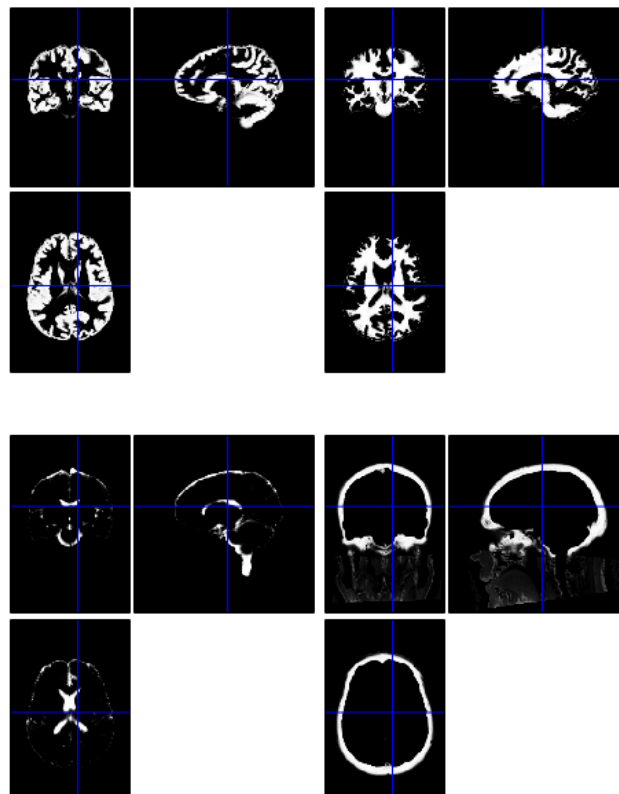


Figura 47. Resultados de la segmentación de T2

En las figuras 46 y 47 se presentan la materia gris (esquina superior izquierda), la materia blanca (esquina

superior derecha), el flujo cerebroespinal (esquina inferior izquierda) y el cráneo (esquina inferior derecha). Las coordenadas de estas imágenes son las nativas de la máquina de donde se hicieron las MRIs T1 y T2. Se observa que en la segmentación del cráneo existen imperfecciones alrededor, aunque segmenta hasta el cuello. Esto también se puede observar en el fluido cerebroespinal. La diferencia más notable en ambas segmentaciones es el cráneo, siendo mucho mejor en los resultados de T2. Tal y como se observa en la figura 48, que muestra una comparativa de la segmentación del cráneo realizada con la T1 y por otro lado con la T2.

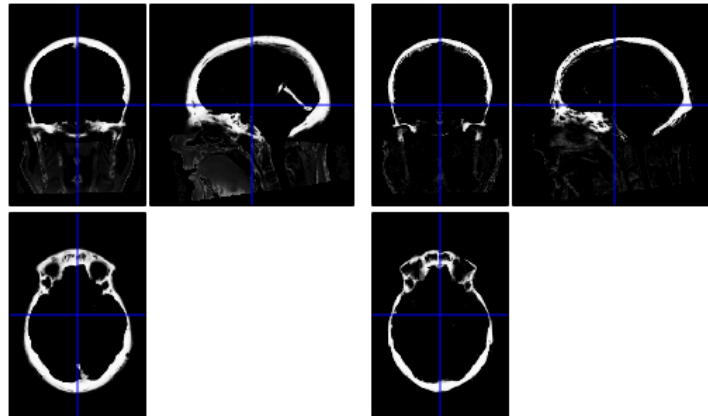


Figura 48. Comparativa de segmentación de cráneo con T1 (derecha), y T2 (izquierda)

SPM12 también puede realizar un suavizado de cada uno de los tejidos que han sido segmentados. En la figura 49 se muestran a la izquierda los suavizados de la materia gris, el fluido cerebroespinal y la piel. En la parte derecha de la figura se muestran los suavizados de la materia blanca y el cráneo.

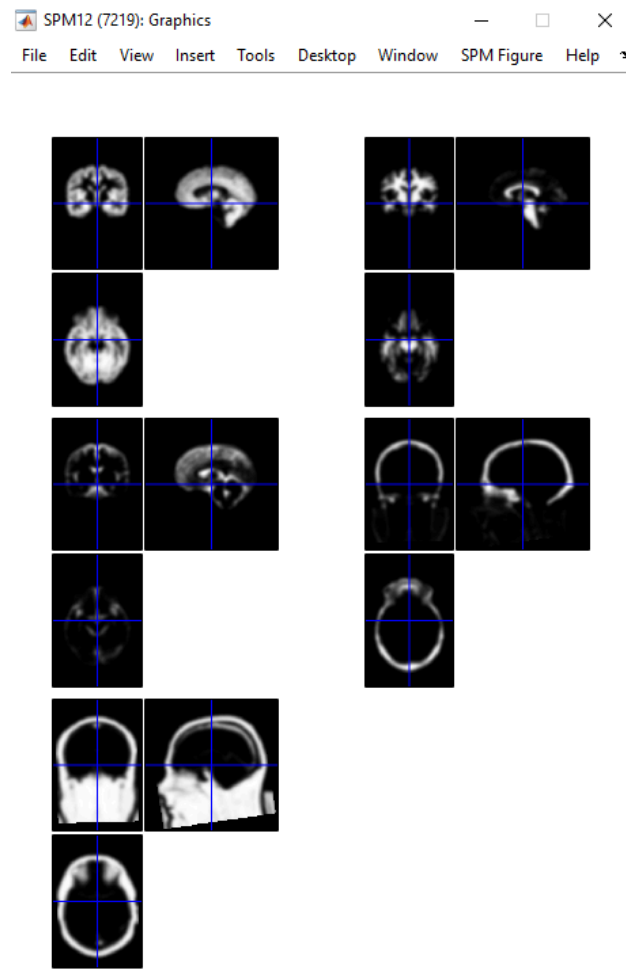


Figura 49. Smoothing de los tejidos segmentados

### 3.3 Resultados de la segmentación realizada con ROAST

ROAST es una herramienta que tiene su propia plantilla llamada eTPM.nii, aunque tiene la opción de añadirle la plantilla que el usuario desee, y está desarrollada en SPM8. En este apartado se muestran los resultados de segmentar las MRIs de estudio con ROAST. Además, se muestra una comparativa de los resultados obtenidos al segmentar las MRIs con ROAST usando la plantilla que tiene SPM8, llamada TPM.nii. En la figura 50 se observa a la izquierda la segmentación realizada con ROAST y a la derecha la segmentación realizada con ROAST utilizando TPM.nii.

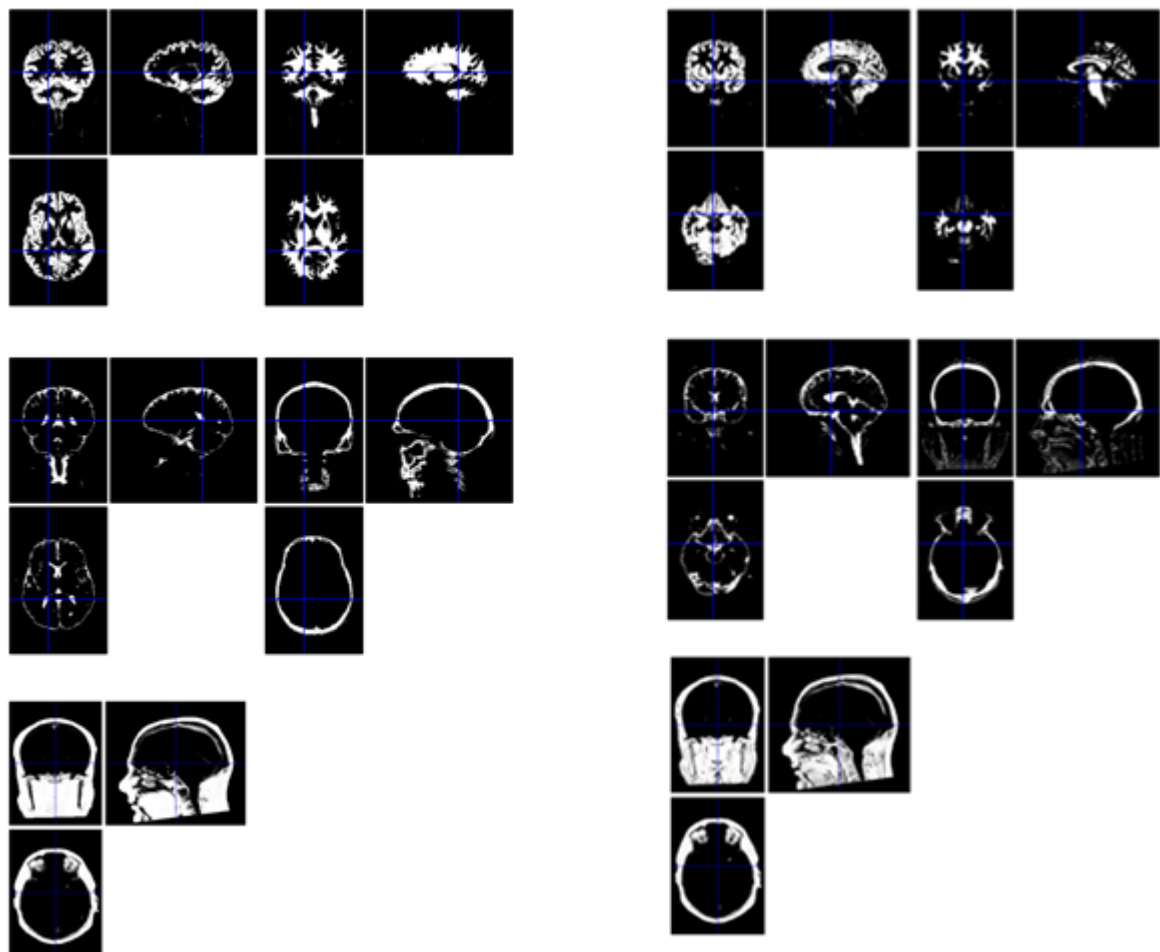


Figura 50. Segmentación con ROAST usando el template de ROAST, eTPM.nii (izquierda), y usando el template de SPM8, TPM.nii (derecha)

Se observa que la parte de la mandíbula y del cuello están mejor segmentadas. Sin embargo, ROAST usa funciones de SPM8, penúltima versión de este software. La segmentación de SPM8 proporciona las bases para la segmentación de SPM12 [87] pero con algunos cambios, como por ejemplo tiene un nuevo template TPM.nii, la segmentación de SPM12 es más precisa que la de SPM8. Dado que SPM12 se ha desarrollado con las bases de SPM8, se ha realizado una mejora incluyendo en el toolbox de ROAST, SPM12 en lugar de SPM8, además de modificar algunas líneas de código de la función `start_seg`. A continuación, se muestra una comparativa de los resultados obtenidos de la segmentación realizada con ROAST, usando el template eTPM.nii, tanto para SPM8 como para SPM12 (figuras 51-55).

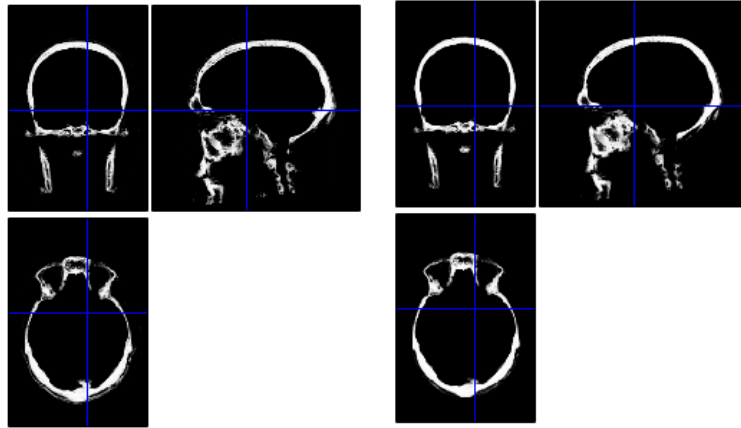


Figura 51. Segmentación del cráneo con ROAST (eTPM), con SPM8 (izquierda) y SPM12 (derecha)

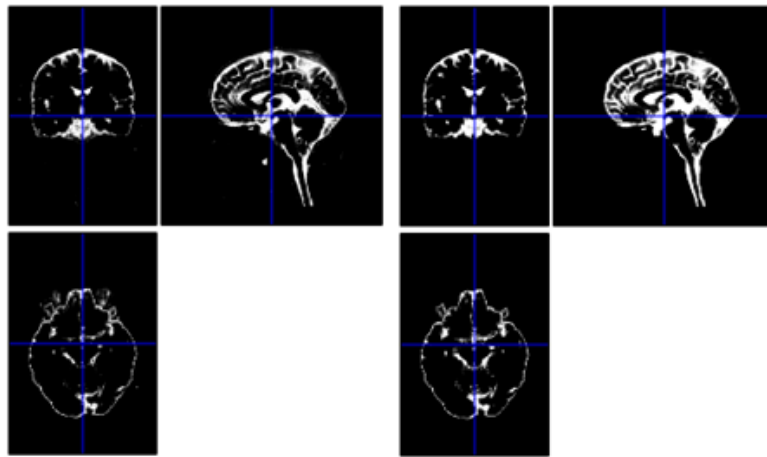


Figura 52. Segmentación del fluido cerebroespinal con ROAST (eTPM), con SPM8 (izquierda) y SPM12 (derecha)

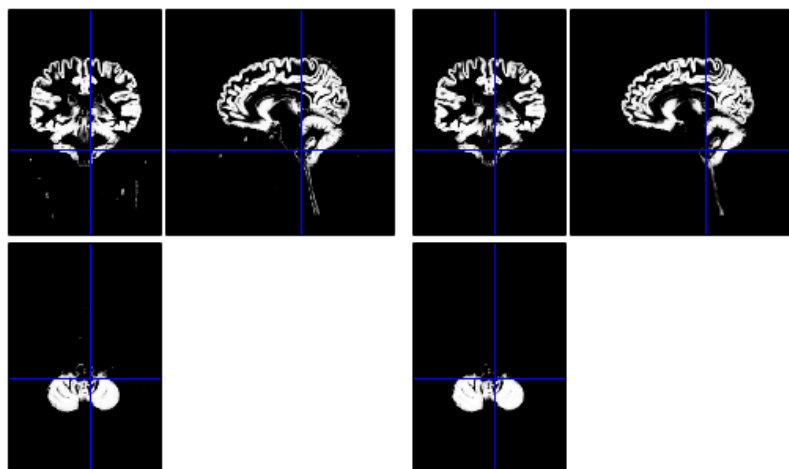


Figura 53. Segmentación de la materia gris con ROAST (eTPM), con SPM8 (izquierda) y SPM12 (derecha)



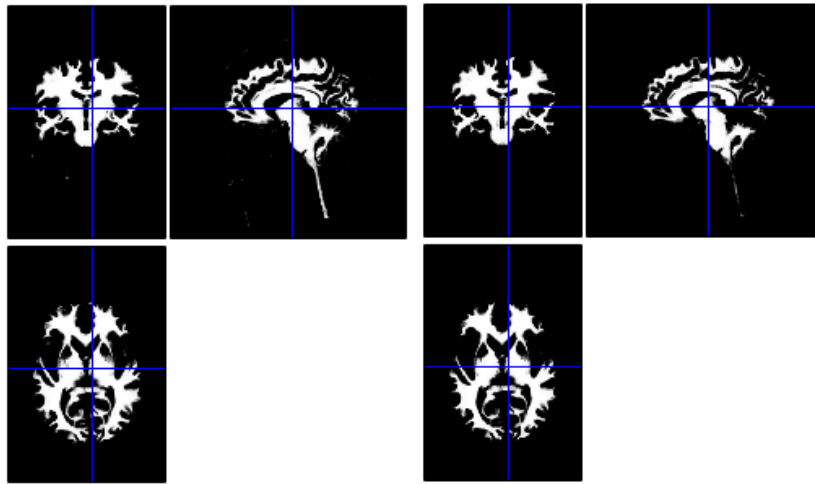


Figura 54. Segmentación de la materia blanca con ROAST (eTPM), con SPM8 (izquierda) y SPM12 (derecha)

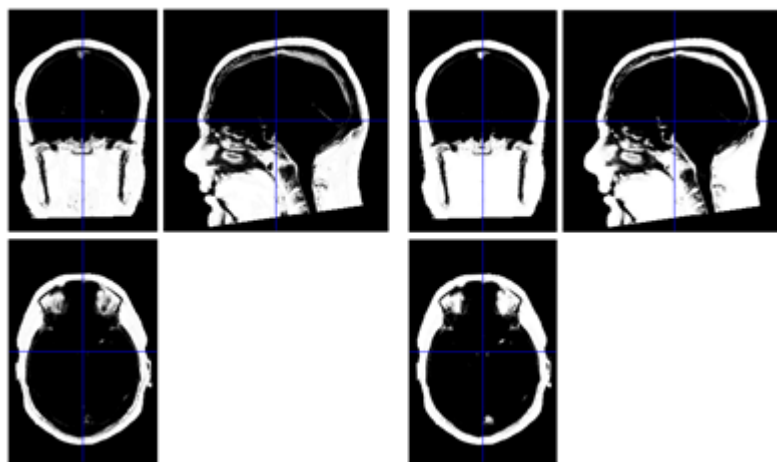


Figura 55. Segmentación de la piel con ROAST (eTPM), con SPM8 (izquierda) y SPM12 (derecha)

En todas las figuras anteriores se puede observar como SPM12 realiza una segmentación más precisa que SPM8 [88]. Por ejemplo, en el cráneo (figura 51), SPM8 detecta como hueso parte de tejido que no lo es, mostrándose en la vista transversal una segmentación menos limpia. Esto ocurre con los demás tejidos, como también se observa con mayor claridad en la materia gris (figura 53).

ROAST también calcula las máscaras de cada uno de los tejidos segmentados, mediante una función llamada *mysegment*. Sin embargo, crea las máscaras con los números 0 y 255. En este trabajo se ha hecho una breve modificación del código para que realice directamente las máscaras binarias (0-1) para poder crear las mallas de este sujeto en apartados posteriores. En la figura 56 se muestran las máscaras realizadas con ROAST tanto para SPM8 como para SPM12. Dado que las mejores segmentaciones se consiguieron utilizando el toolbox ROAST, con el template que trae por defecto, eTPM, y añadiendo SPM12 en lugar de SPM8, como era de esperar las mejores máscaras también se han obtenido con SPM12.

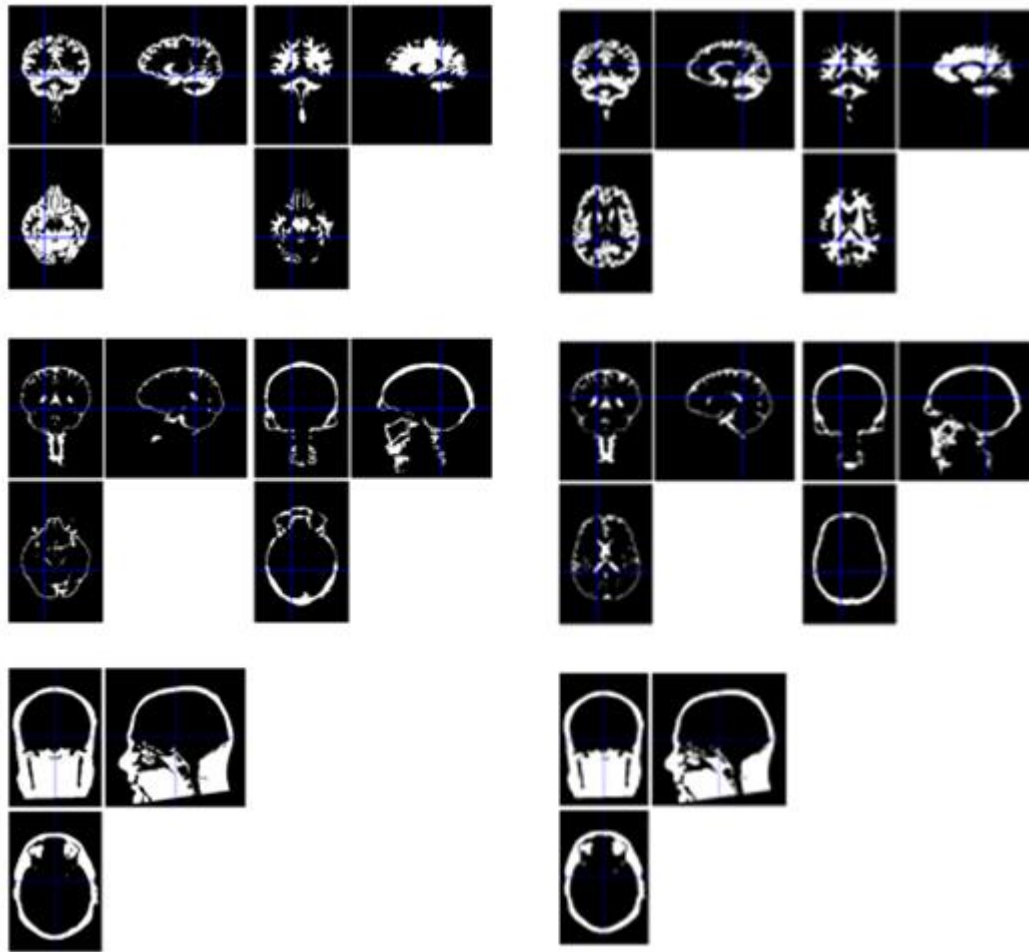


Figura 56. Máscaras binarias con ROAST, SPM8 (izquierda), SPM12 (derecha)

### 3.4 Resultados de la segmentación realizada con SIMNIBS

En este apartado solo se presentarán los resultados de las segmentaciones con SIMNIBS, para las MRIs T1 y T2 del mismo sujeto. Al final la segmentación, se pueden representar las superficies reconstruidas superpuestas sobre la MRI usando FreeSurfer. Para ello basta con usar el siguiente comando: `mri2mesh -c almi5`. Esto abre la herramienta FreeSurfer y hace una comprobación de la calidad de las segmentaciones, luego fusiona los archivos stl y los pasa al formato de FreeSurfer.

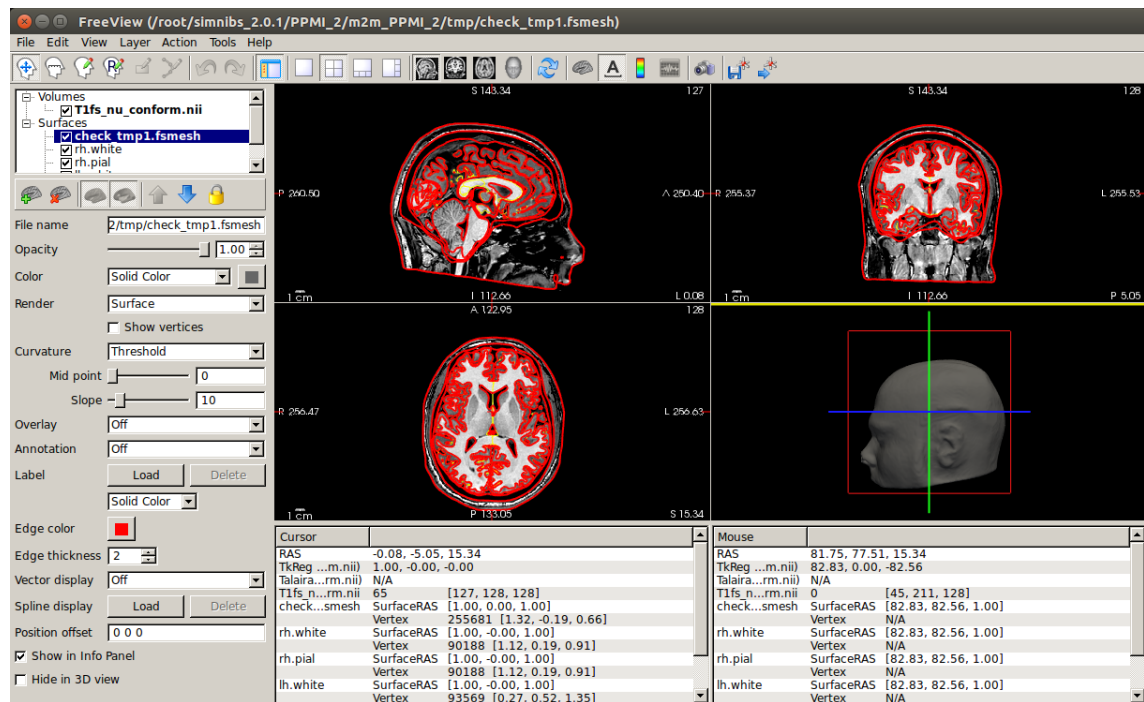


Figura 57. Segmentación de FreeSurfer, para reconstrucción de malla volumétrica

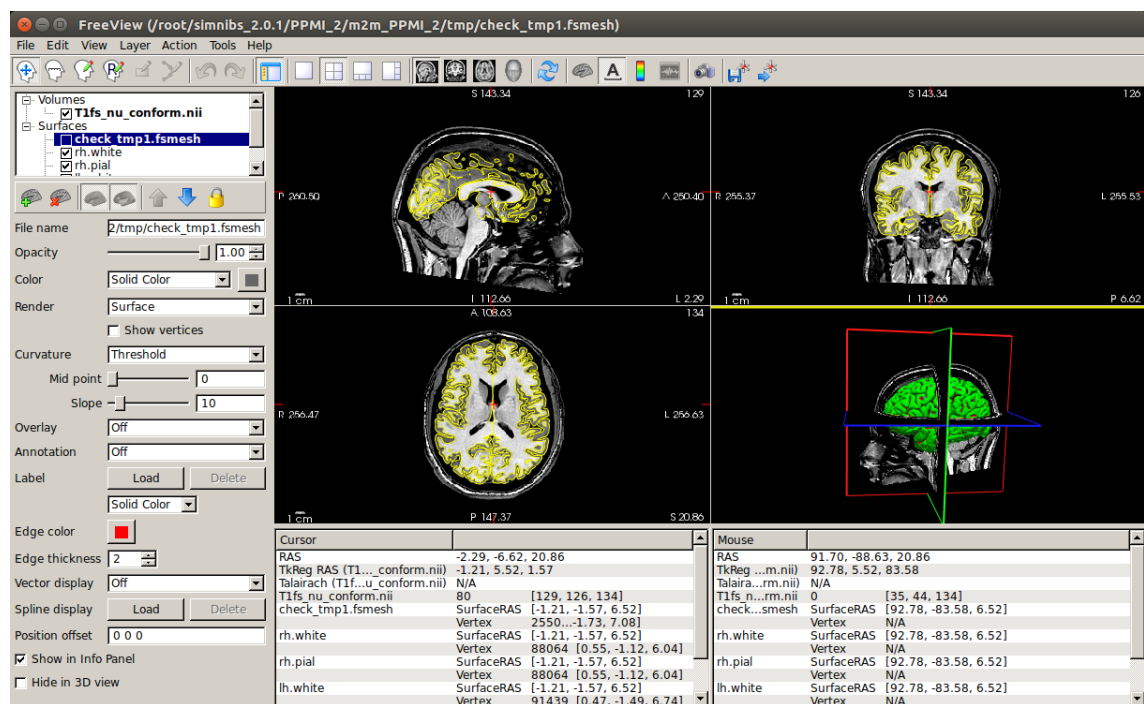


Figura 58. Segmentación de FreeSurfer de materia blanca y gris

En las interfaces gráficas de las figuras 57 y 58 se pueden observar cuatro vistas: vista sagital, coronal, trasveral y una representación en 3 dimensiones. La línea roja indica las superficies usadas para la reconstrucción de la malla volumétrica. La línea amarilla indica las superficies de materia blanca y gris creadas por FreeSurfer. Además, en la parte inferior de la pantalla se muestran las coordenadas por las que el ratón pasa y las coordenadas en las que el cursor fijo se sitúa. También aparece la orientación de la imagen, RAS, el sistema de coordenadas usado que es Talairach [89] [90].

En las figuras 59-62 se muestran las máscaras obtenidas a partir de la segmentación realizada representadas con NIFTI toolbox. Cabe destacar que SIMNIBS también segmenta el cerebelo y los ventrículos, sin embargo, no se segmenta el cuello y, además, el cráneo es muy poco realista ya que no presenta las cavidades en los ojos. Por el contrario, realiza una buena segmentación de la materia blanca y gris.

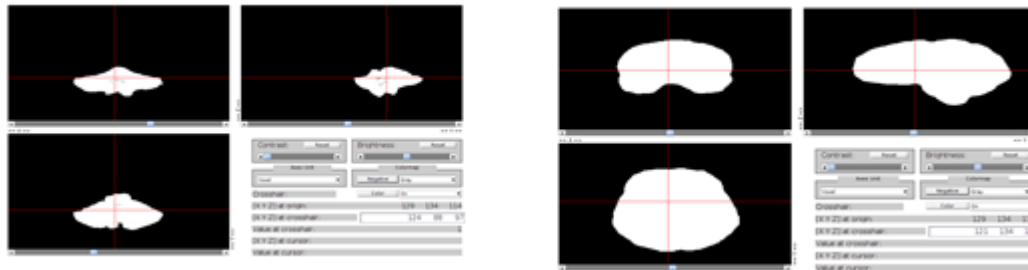


Figura 59. Segmentación del cerebelo (izquierda) y líquido cerebroespinal (derecha) representada con NIFTI toolbox

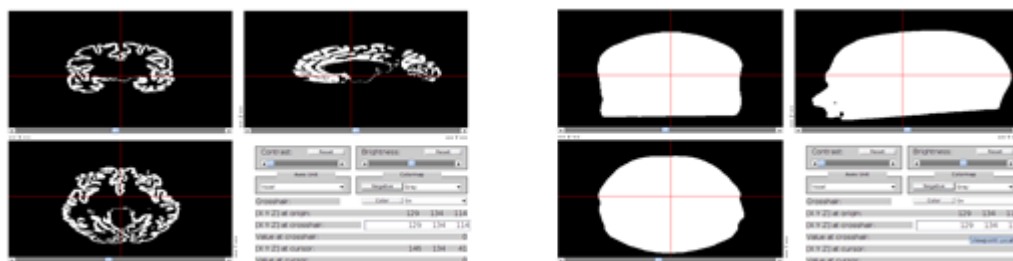


Figura 60. Segmentación de la materia gris (izquierda) y de la piel (derecha) representada con NIFTI toolbox

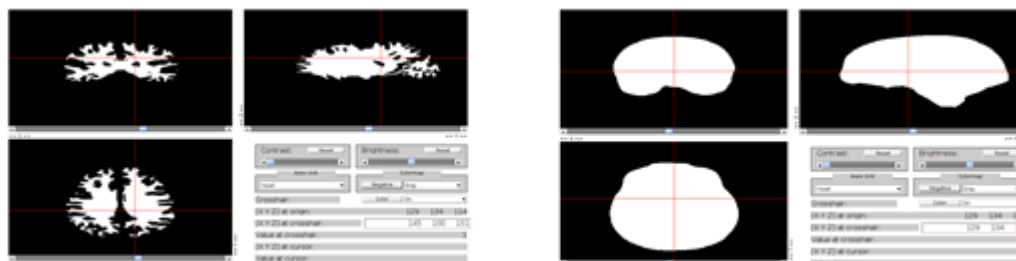


Figura 61. Segmentación de la materia blanca (izquierda) y cráneo (derecha) representada con NIFTI toolbox

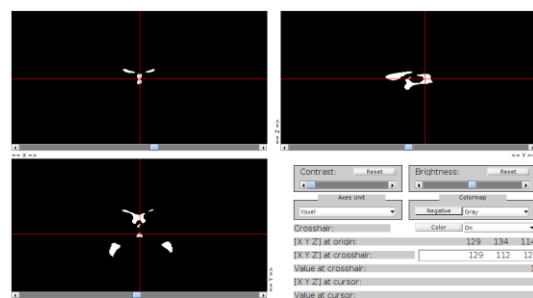


Figura 62. Segmentación de ventrículos representada con NIFTI toolbox

### 3.5 Resultados del mallado superficial

#### 3.5.1 Resultados del mallado superficial con Binsurface

En primer lugar, se muestran en las figuras 63-67 los resultados obtenidos de hacer el mallado superficial a las máscaras resultantes de las segmentaciones realizadas con ROAST, haciendo una comparativa del mallado a partir de las máscaras que se obtuvieron segmentando con ROAST pero sustituyendo SPM8 por SPM12. Cabe destacar que la plantilla usada es la que contiene ROAST, eTPM.nii, con la que se obtuvieron mejores resultados en los apartados anteriores.

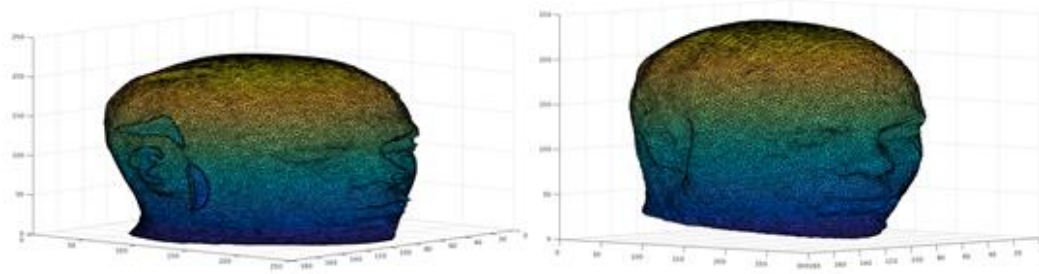


Figura 63. Malla superficial de la piel creada con binsurface a partir de segmentación con SPM8 (izquierda) y SPM12 (derecha)

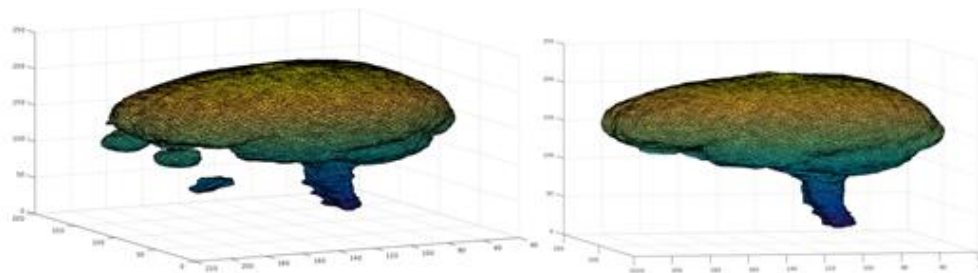


Figura 64. Malla superficial del líquido cerebroespinal creada con binsurface a partir de segmentación con SPM8 (izquierda) y SPM12 (derecha)

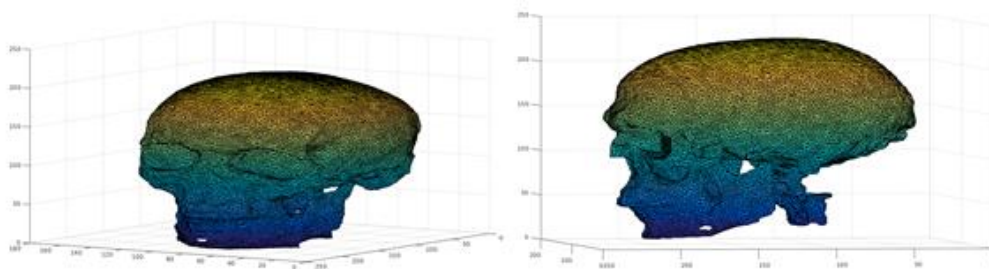


Figura 65. Malla superficial del cráneo creada con binsurface a partir de segmentación con SPM8 (izquierda) y SPM12 (derecha)



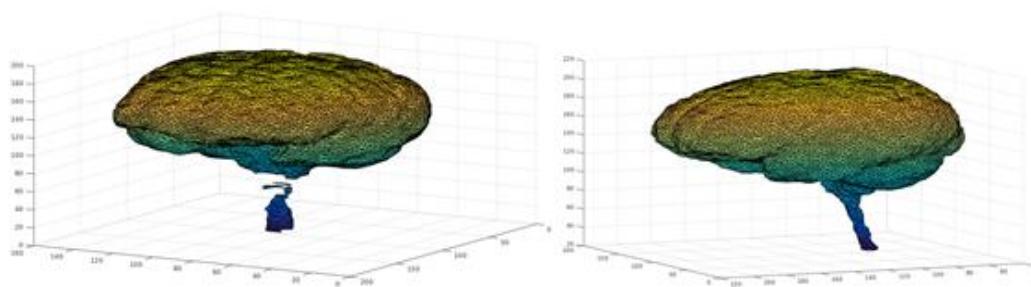


Figura 66. Malla superficial de la materia gris creada con binsurface a partir de segmentación con SPM8 (izquierda) y SPM12 (derecha)

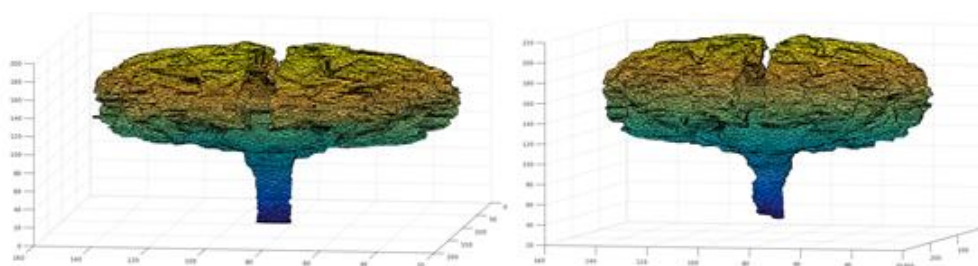


Figura 67. Malla superficial de la materia blanca creada con binsurface a partir de segmentación con SPM8 (izquierda) y SPM12 (derecha)

Como era de esperar, las mallas superficiales a partir de las máscaras creadas con SPM12 dan mejores resultados que las creadas a partir de SPM8. En la malla de la piel se observa que se necesitaría aumentar la resolución (parámetro `fillholes3D`) para conseguir cerrar ese agujero. Sin embargo, Al aumentar la resolución, influye en que los huecos de los ojos se cierran aun más y sería una solución aun menos realista. Cabe destacar, que al segmentar con ROAST haciendo uso de SPM8 se debió identificar como fluido cerebroespinal, partes de la cabeza que no lo eran, como lo ojos. Justo lo contrario ocurre con la segmentación de la materia gris, como se muestra en su correspondiente malla superficial.

En segundo lugar, se muestran en las figuras 68-71 los resultados obtenidos con las máscaras que calcula SIMNIBS, usando el método de `binsurface`. La reconstrucción de la malla superficial con las máscaras de SIMNIBS son muy buenas. Sin embargo, debido a la segmentación no muy realista del cráneo, el mallado del mismo es bastante artificial, incluso puede llegar a confundirse con el fluido cerebroespinal. Además, el mallado de la piel no llega a reconstruir el cuello. Por otro lado, el mallado de la materia gris y blanca es bastante bueno con este método y gracias a las segmentaciones hechas con SIMNIBS.

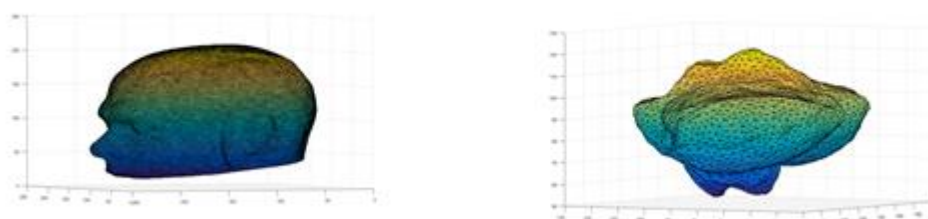


Figura 68. Malla superficial de la piel (izquierda) y del cerebelo (derecha) creada con binsurface a partir de la segmentación de SIMNIBS



Figura 69. Malla superficial del líquido cerebroespinal (izquierda) y de la materia gris (derecha) creada con binsurface a partir de la segmentación de SIMNIBS

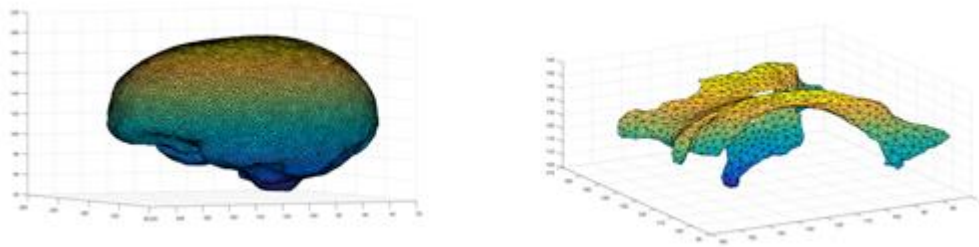


Figura 70. Malla superficial del cráneo (izquierda) y ventrículos (derecha) creada con binsurface a partir de la segmentación de SIMNIBS

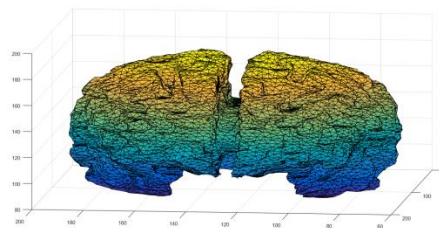


Figura 71. Malla superficial de la materia blanca creada con binsurface a partir de la segmentación de SIMNIBS

### 3.5.2 Resultados del mallado superficial con Vol2surf

En este subapartado, en primer lugar, se muestran los resultados del mallado superficial usando la opción de vol2surf a los tejidos de la piel y el cráneo. Solo se mostrarán los resultados usando el método cgalsurf [91] [92], ya que el método de simplify utiliza binsurface y daría los mismos resultados que en el apartado anterior.

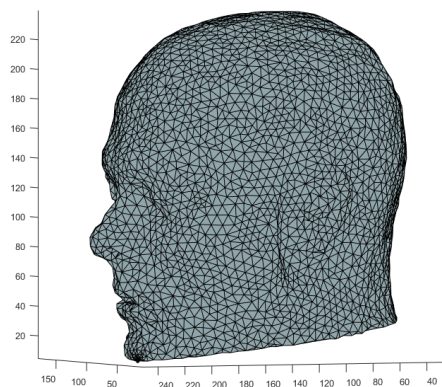


Figura 72. Malla superficial de la piel con método cgalsurf

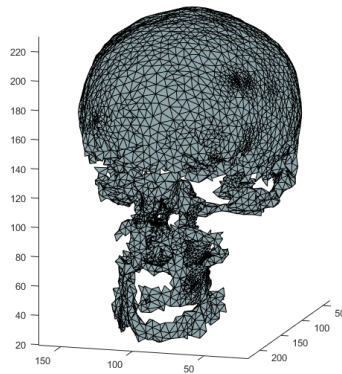


Figura 73 Malla superficial del cráneo con método `cgalsurf`

Se ha utilizado este método con un tamaño del triángulo de 5. Se observa en la figura 73 que este método deja unos surcos en la malla y el mallado del cráneo es bastante deficiente ya que parece que no ha conseguido reconstruirlo por completo. Por ello, se ha decrementado el tamaño de los triángulos a 2, para comprobar si existe alguna mejora.

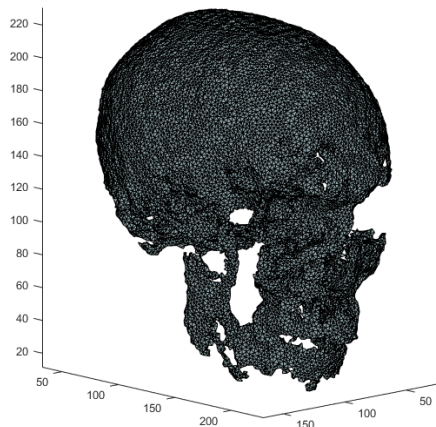


Figura 74. Malla superficial del cráneo con método `cgalsurf`. Menor tamaño de triángulos

En la figura 74 se observa que al decrementar el tamaño de los triángulos se mejora la malla superficial, pero aun así los resultados no son tan buenos como con la opción de `binsurface`.

Por otro lado, decir que ROAST crea la malla superficial con el método `cgalsurf` pero no se obtienen mallas de muy buena calidad, como se puede observar en la figura 75.





Figura 75. Mallado del líquido cerebroespinal mediante ROAST.

### 3.6 Resultados del mallado volumétrico con Surf2mesh

Una vez obtenidas las mallas superficiales se puede proceder a crear las mallas volumétricas. En este subapartado se mostrarán los resultados de mallas volumétricas a partir de las mallas superficiales de SIMNIBS.

En primer lugar, se muestra en la figura 76 la fusión de todos los tejidos mediante `mergesurf`. Para ello se han cargado todos los archivos `stl` que se corresponden con las segmentaciones de la piel, el cráneo, la materia blanca y gris, los ventrículos, el fluido cerebroespinal y el cerebelo. Cabe destacar que `mergesurf` funciona solo con matrices binarias y se obtienen dos matrices nuevas que contienen los nodos y las caras de todos los tejidos juntos.

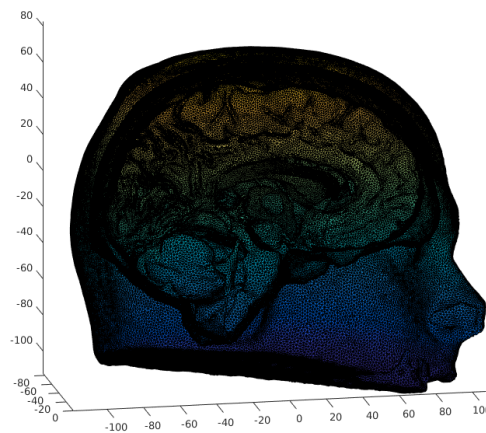


Figura 76. Fusión de los tejidos segmentados con SIMNIBS

En segundo lugar, se muestra en la figura 77 la malla volumétrica reconstruida con `surf2mesh` utilizando el método `tetgen`. Para ello se necesita un punto interior correspondiente a cada tejido y las matrices de nodos y caras obtenidas con `mergesurf`. `surf2mesh` da como resultado las matrices de nodos, caras y elementos, y es en esta última donde se definen los vértices de los tetraedros. Para poder visualizar los tetraedros es preciso representar las matrices de nodos frente a los tetraedros, dividiendo la cabeza del sujeto por la mitad.

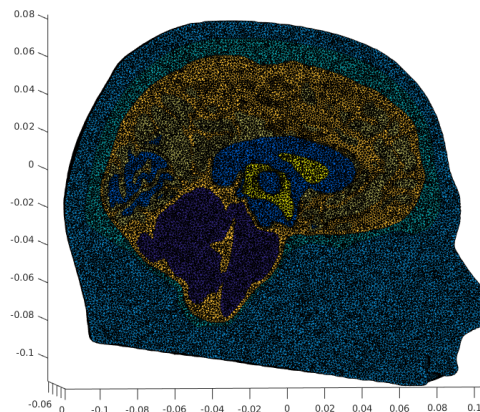


Figura 77. Malla de tetraedros a partir de los tejidos segmentados con SIMNIBS

Cabe destacar que las segmentaciones realizadas con SPM tienen muchas intersecciones y esto impide poder realizar la malla de tetraedros con los tejidos segmentados mediante SPM12.

### 3.7 Resultados de la colocación de electrodos

Tras los resultados obtenidos con las herramientas *Metch* y *Mesh2EEG*, se ha realizado un código para la colocación de los electrodos y la creación de los mismos, utilizando funciones de ambas herramientas y de *iso2mesh*.

Para la colocación de las ubicaciones de los electrodos se ha utilizado la función *ComputeEEGPos* de *Mesh2EEG* a la que, originalmente, se le pasa como argumentos de entrada los cuatro puntos de referencia (Nz, Iz, T3 y T4) [93], las caras de los triángulos, los nodos y el sistema EEG. Como se ha explicado antes, los puntos de referencia se introducían manualmente en el código a partir de la representación de la malla superficial. En este trabajo se ha realizado un algoritmo [94] para que estos cuatro puntos se almacenen en variables, llamadas Nz, Iz, T3 y T4, automáticamente al marcarlas sobre la superficie. Por otro lado, la función *ComputeEEGPos* contempla cuatro casos para la ubicación de los electrodos que son el sistema 10/05, el sistema 10/10, el sistema 10/20 y el sistema 10/10 [95] convertido al sistema NIRS. Otra mejora incorporada es añadir un caso más que consiste en, partiendo de las localizaciones de los electrodos del sistema 10/20, elegir qué electrodo colocar. Además, se ha creado una función para que, a través de un menú, se elija alguna de estas cinco opciones y devuelva el número de la opción elegida para que forme parte de los argumentos de entrada de *ComputeEEGPos*.

Una vez ubicadas todas las posiciones de los electrodos se procede a la reconstrucción de los mismos. En primer lugar, se ha localizado un punto interior a la cabeza del sujeto calculando la media de todos los centros de los triángulos que componen la malla superficial, con ayuda de la función *meshcentroid* de *iso2mesh*. A continuación, se han dibujado rectas desde ese punto interior hasta cada una de las ubicaciones de los electrodos, que luego se han prolongado hacia fuera de la cabeza. Por lo que hasta ahora se tienen tres tipos de puntos: el que se encuentra en el punto medio interior de la cabeza (denominado *mediaCentroid*), el que se encuentra en la superficie (denominado *EEGPos*) y el que se encuentra en el extremo de la recta prolongada (denominado *pext*). Ya que esa recta no es normal a la superficie, se busca un punto entre *EEGPos* y *pext* que sea muy cercano a la superficie, es decir, a *EEGPos* (denominado *Esp*). Este punto definirá el espesor del electrodo). Con ese nuevo punto se calcula la proyección normal del mismo con la superficie, con la función *proj2mesh* de la herramienta *Metch*. Esto da lugar a otro nuevo punto sobre la superficie y muy cercano a *EEGPos* (denominado *points\_after\_proj*). Como la recta que une los puntos *Esp* y *points\_after\_proj* si es normal a la superficie, se puede proceder a la reconstrucción de los electrodos.

Los electrodos se han creado con la función *meshcylinder* de *iso2mesh*. Tienen forma de cilindro, cuyo espesor está definido por la longitud de su eje central marcado por los puntos *Esp* y *points\_after\_proj*, y el radio, que se le puede pasar como argumento de entrada. Por último se extrae la superficie de los cilindros

con la función `surftri` y se representa con `trisurf`. En la figura 78 se muestra el resultado de la reconstrucción de la superficie de triángulos de la piel del sujeto, junto con los electrodos colocados sobre la malla ubicados según el sistema 10/20.

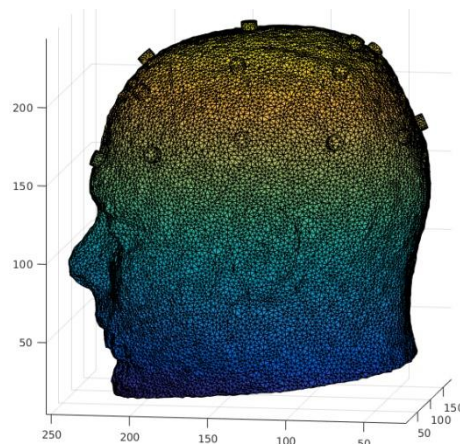


Figura 78. Colocación de electrodos con algoritmo mejorado

### 3.8 Flujo de trabajo para la elaboración de un modelo computacional del cerebro

El proceso que se propone se ha realizado en Matlab, utilizando las herramientas explicadas anteriormente, y seleccionando las que mejores resultados han obtenido. En primer lugar, se realiza la segmentación y la posterior creación de las máscaras, haciendo uso de la herramienta ROAST con SPM12 en lugar de SPM8, y con la plantilla que incorpora ROAST, `eTPM.nii`. Las máscaras creadas son binarias con niveles entre 0 y 1, en lugar de 0 y 255 para, posteriormente, introducir la máscara de la piel en `binsurface` para la reconstrucción del mallado superficial. Por último, la colocación de electrodos se consigue con el algoritmo mejorado en este trabajo.

Es recomendable que sea ejecutado en linux. El directorio principal se llama `workflow` y dentro existen multitud de funciones para la elaboración del flujo de trabajo. Además, se encontrarán todas las carpetas necesarias para ello. Si el usuario dispone de las MRIs en formato DICOM, se aconseja que sean introducidas dentro de las carpetas `T1_dicom` y `T2_dicom`, según corresponda. Si por el contrario dispone de las MRIs en formato NIFTI, se aconseja que las almacene en las carpetas `T1_nii` y `T2_nii`, de igual forma.

El primer paso es ejecutar el script principal llamado `workflow`. En ese momento se abrirá un cuadro de diálogo en el que el usuario deberá indicar la ruta del directorio donde se encuentra ese script (figuras 79 y 80).

```
*** Select the workflow directory ***
```

Figura 79. Mensaje en consola de Matlab para introducir la ruta del directorio principal

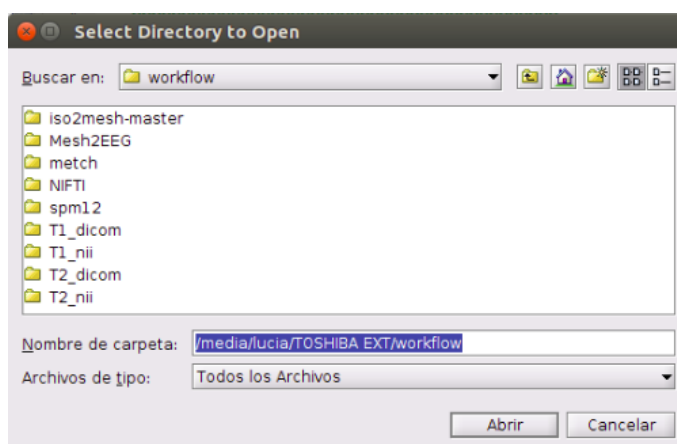


Figura 80. Cuadro de diálogo para la selección del directorio principal

El programa se iniciará con la conversión de los archivos DICOM, si los hubiera. En esta descripción se plantea el ejemplo de que solo existen archivos tipo DICOM para la T2. Por lo que se convierten al formato NIFTI con destino la carpeta T2\_nii. Una vez que ambas imágenes se encuentran en las carpetas T1\_nii y T2\_nii se procede a la descompresión de ambas (figura 81).

```

*** Select the workflow directory ***
*** DICOM files dont exist for T1 ***
Xiangrui Li's dicm2nii (feedback to xiangrui.li@gmail.com)
Validating 171 files (Philips) ...
Converting 1 series into .nii.gz: subject CORTES DOMINGUEZ^JESUS
  x3D_T2      170
Elapsed time by dicm2nii is 2.3 seconds

*** Convert DICOM files to NIFTI in /T2_nii ***
*** The dicom files are uncompressed ***

SPM12: spm_coreg (v6435)                                22:27:57 - 29/06/2018
=====
Completed                                           :                22:28:07 - 29/06/2018

SPM12: spm_reslice (v7141)                             22:28:07 - 29/06/2018
=====
Completed                                           :                22:28:17 - 29/06/2018

```

Figura 81. Conversión de archivos DICOM a archivos NIFTI y descompresión de archivos NIFTI

Automáticamente, las imágenes serán igualadas en dimensiones y alineadas. Y se creará el correspondiente archivo rT2.nii. A continuación, se procede a reorientar las imágenes, según se muestra en la figura 82, poniendo el centro de coordenadas en la comisura anterior. Se abre primero la MRI T1 y se selecciona dicho punto para luego guardar su orientación. Se copian los vóxeles que marcan el origen y se pegan en la MRI T2.

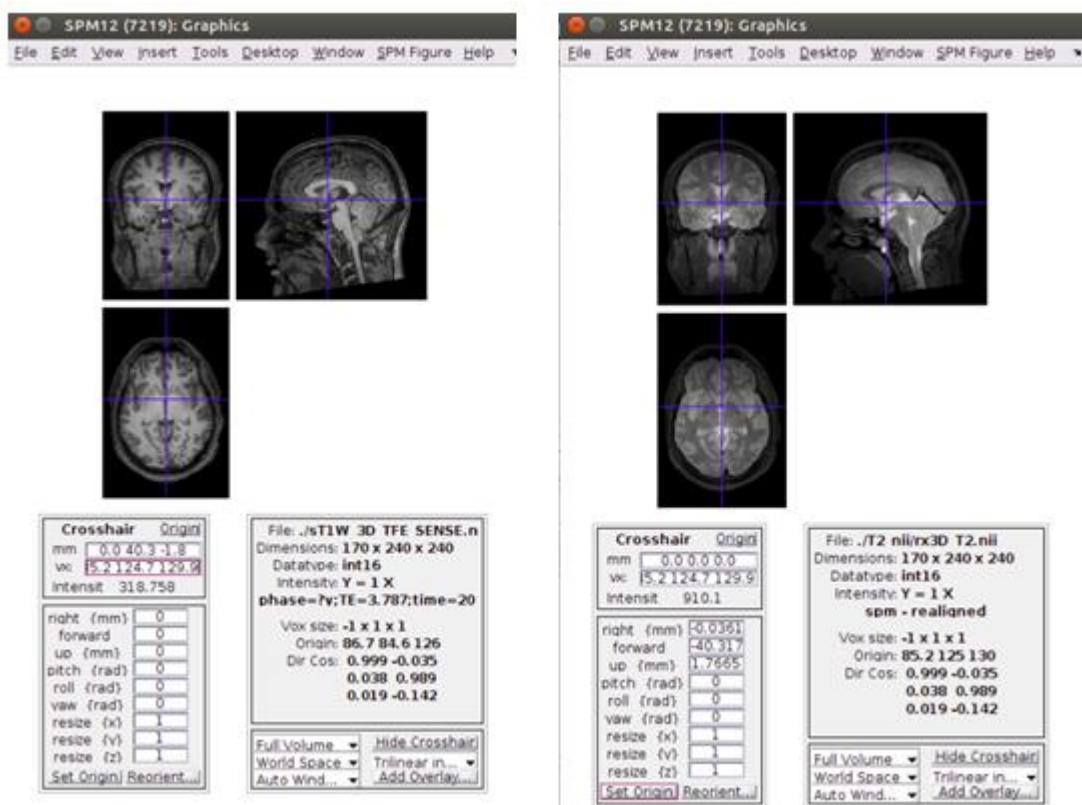


Figura 82. Reorientación de la imagen T1 (izquierda) y T2 (derecha)

El siguiente paso es la segmentación de las imágenes y la creación de las máscaras binarias de cada tejido, que se realiza de forma automática (véanse las figuras 83-85). Cuando ha terminado, se selecciona la carpeta donde se encuentran todas las máscaras binarias y se selecciona la máscara de la piel.

```
loading data...
smoothing masks...
creating binary masks...
fixing CSF continuity...
removing disconnected voxels...
generating and labeling empty voxels...
removing outside air...
Seleccione el directorio donde se encuentra la máscara binaria de la piel...
```

Figura 83. Creación de máscaras binarias e indicación para seleccionar el directorio donde encuentran

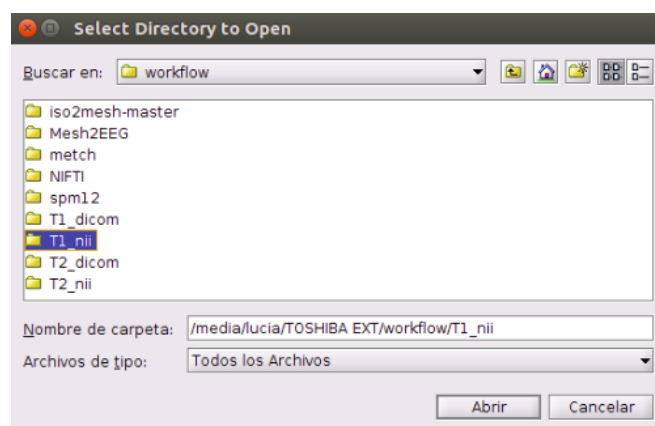


Figura 84. Cuadro de diálogo para la selección del directorio de las máscaras

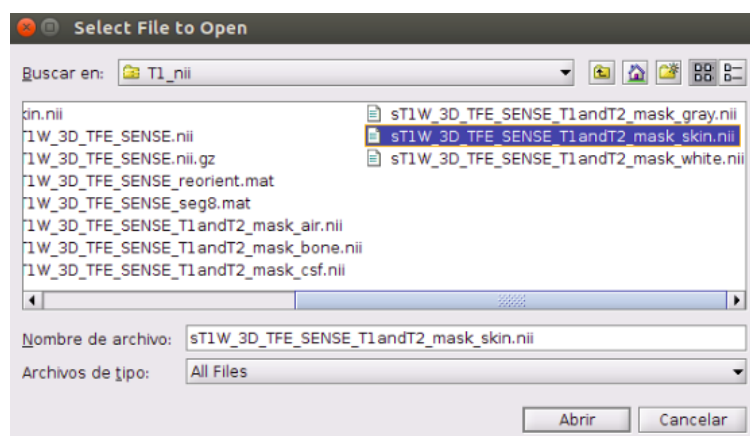


Figura 85. Cuadro de diálogo para seleccionar la máscara de la piel

Automáticamente se crearán los nodos y las caras de los triángulos que construirán la malla superficial de la piel del sujeto. Al finalizar, se procede a la colocación de los electrodos. Para ello, según se muestra en la figura 86, se inserta el radio y el espesor de los electrodos en forma de cilindro.

```
Edges processed: 203560
Edges collapsed: 203136

Edges not collapsed due to topological constraints: 423
Edge not collapsed due to cost computation constraints: 0
Edge not collapsed due to placement computation constraints: 0

Finished...
609408 edges removed.
67710 final edges.
Enter the cylinder radius: 8
*** Press enter to continue ***
Enter the thickness of the cylinder: 4
*** Press enter to continue ***
```

Figura 86. Introducción del radio y espesor del cilindro

De forma automática, se presenta la cabeza del sujeto, sobre la que se deberán marcar los cuatro puntos de



referencia (Nz, Iz, T3, T4) según va indicando el menú. La figura 87 muestra un ejemplo en el que se selecciona el punto Nasion.

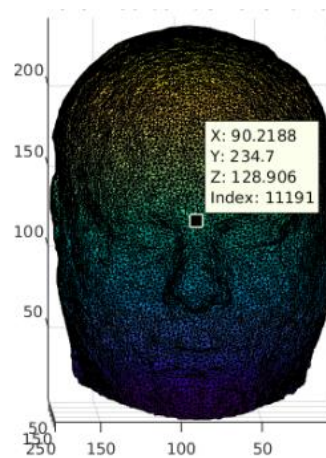


Figura 87. Localización del punto de referencia Nasion

Al finalizar, se elige cuál de las cinco opciones posible se desea para la colocación de los electrodos. En este ejemplo (figuras 88-89) se ha seleccionado la opción cinco para colocar dos electrodos específicos, el C3 y el Fp2.

```
Loading ..97%
Loading ..99%
INFO- Loaded 22572 vertices and 45140 faces.
Saving Manifold Oriented Triangulation ...
(*)First load the mesh and point cloud. Hit Enter to continue...
(*)Create 4 mapping pairs to initialize the mapping. Hit Enter to continue...
(*)Click on Nasion
(*)Click on Iz
(*)Click on T3(left ear)
(*)Click on T4(right ear)
(*)Hit Enter to continue...
```

Figura 88. Puntos de referencias insertados

```
(*)Elige un sistema de coordenadas:
Opción 1: Sistema 1005
Opción 2: Sistema 1010
Opción 3: Sistema 1020
Opción 4: Sistema 1010 NIRS
Opción 5: Elija el electrodo que quiera colocar|
Ingrese opción:5
Ejemplo de formato: {'Fp1',1,'P4',-1}
{'C3',1,'Fp2',-1}|
```

Figura 89. Selección de electrodos a colocar sobre la cabeza

Por último, se reconstruyen los electrodos especificados en el menú sobre la malla de la cabeza, según se muestra en la figura 90.

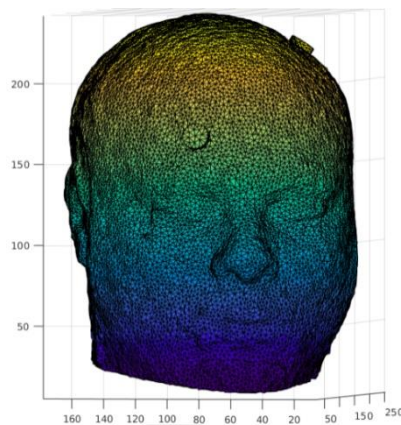


Figura 90. Electrodo sobre la malla superficial de triángulos

### 3.9 Aplicación al software Brainsight

Brainsight se creó en el año 2000 para desarrollar nuevas herramientas que ayudaran a la investigación de la neurociencia y, actualmente, existen cientos de laboratorios en todo el mundo que utilizan este software. Es importante tener en cuenta que Brainsight está dedicado a la investigación científica y no a aplicaciones médicas.

La aplicación de Brainsight a TMS fue el primer navegador basado en imágenes disponible comercialmente dedicado a la colocación de TMS, que se ha convertido con el paso de los años en el neuronavegador más popular para TMS. Se utiliza en más de 500 laboratorios en todo el mundo.

Con más de 15 años de comentarios de cientos de investigadores, el sistema de navegación Brainsight TMS ha evolucionado continuamente como el centro neurológico universal a través de la integración con sistemas de neuroestimulación y neuroimagen [96]. Los beneficios clave incluyen:

- Compatibilidad universal, trabajando con cualquier sistema TMS de cualquier fabricante.
- Flujo de trabajo de software intuitivo, desde la configuración hasta la realización de TMS en solo unos minutos.
- Selección flexible de objetivos, basada en: anatomía, coordenadas estandarizadas (MNI [97], Talairach [98]), superposiciones múltiples (CT, EEG, fMRI, MEP, NIRS, PET, etc.), ubicaciones de bobina grabadas de sesiones anteriores.
- Grabaciones detalladas almacenadas para cada pulso TMS.
- Múltiples formatos de archivos de imagen compatible con: DICOM, MINC, Analyze, PAR/REC, NIFTI, Siemens .ima, BrainVoyager, VMR, VMP.
- Datos totalmente exportables para análisis fuera de línea.
- Herramienta de pintura de región de interés (ROI) para la reconstrucción en 3D o para resaltar una estructura, que se puede exportar a otro software como un archivo NIFTI.
- Reconstrucción cerebral 3D curvilínea automática
- Diseños, vistas y reconstrucciones personalizadas para adaptarse a todos los usuarios y requisitos.
- Creación de objetivos utilizando marcadores o trayectorias.

El objetivo en esta parte del trabajo ha sido localizar cualquier punto en la cabeza de un sujeto real a partir de las coordenadas de un modelo computacional personalizado, en función de la MRI de un individuo, partiendo del flujo de trabajo desarrollado en la sección anterior.

Brainsight dispone de dos formatos distintos para la localización de la bobina sobre el sujeto que son, el formato



Brainsight propio del software y el formato NIFTI, el cual ha sido clave para resolver el problema planteado.

En primer lugar, se extrajo la MRI del software Brainsight que estaba siendo utilizada para los estudios de TMS, en formato NIFTI. Se procedió a la segmentación y creación de las máscaras de esta imagen con la herramienta ROAST empleando SPM12. Dado que solo se dispone de una MRI, no es necesario realizar el correregistro ni la alineación de imágenes [99]. Además, es muy importante no reorientar la imagen, ya que esto define un nuevo origen de coordenadas, y provocaría que las coordenadas de la MRI original del software Brainsight no coincidan con las coordenadas de la imagen segmentada.

En segundo lugar, se realiza la reconstrucción de la máscara de la piel, creando los nodos y triángulos de la malla superficial mediante la función `binsurface` de `iso2mesh`. Las coordenadas en las que se encuentra la superficie triangular de la cabeza del sujeto coinciden con las coordenadas de la MRI segmentada y, por tanto, de la original, con la salvedad de que en la malla superficial estas coordenadas se encuentran definidas en unidades de vóxel. Por lo que si se selecciona un punto cualquiera sobre la superficie mallada del sujeto debe corresponder tanto en el tejido segmentado de la piel, como en la MRI original.

La ventaja que tiene representar la MRI con la herramienta NIFTI es que realiza la conversión de vóxel a milímetros. Entonces, a partir de las coordenadas en vóxel de la cabeza mallada se pueden obtener las coordenadas en milímetros de un punto determinado. Dado que el programa Brainsight es compatible con el formato NIFTI, ese punto debería coincidir dentro de este software.

Por otro lado, Brainsight dispone de un sistema de neuronavegación similar al sistema GPS, donde el satélite sería reemplazado por un sensor de posición, normalmente una cámara óptica. La antena del GPS que recibe sus señales es reemplazada por un “tracker”, que consiste en un objeto triangular pequeño con tres o más esferas reflectoras. Este objeto se encuentra en la cabeza del sujeto y en la bobina. El mapa del GPS sería la MRI del sujeto. El software de navegación comunica con el sensor de posición con los trackers. Para mapear el espacio de la MRI con el espacio real donde se encuentra el sujeto, se deben obtener algunos puntos de referencia en la MRI como el Nasion, la nariz y los tragus. Luego se deben buscar estos mismos puntos en la cabeza real del sujeto para que con ayuda de una matriz de registro se realice la correspondencia del espacio MRI con el mundo real, para así mapear la posición de la bobina de TMS.

Una vez hecho esto, se puede localizar el punto que fue definido sobre el modelo computacional realizado gracias a la segmentación de la MRI original [100], en la cabeza del sujeto real. Cabe destacar, que este flujo de trabajo se puede realizar a la inversa. Se puede ubicar una zona de estimulación sobre la cabeza del sujeto en el mundo real y conseguir localizar esa zona en el modelo computacional.

A continuación, se muestran varias imágenes que explican el proceso de conversión de coordenadas de un espacio a otro. La figura 91 muestra la cabeza mallada de la piel del sujeto con un punto selecciona a unos pocos centímetros de la oreja izquierda.

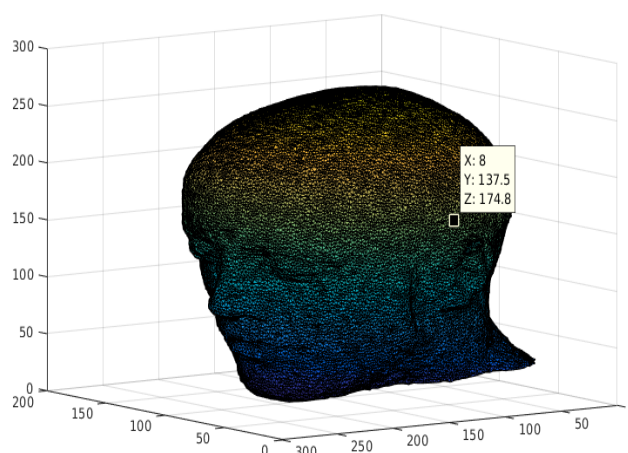


Figura 91. Coordenada en vóxel de un punto sobre la malla superficial

En las figuras 92 y 93 se muestra la imagen original MRI con esas coordenadas de vóxel localizadas y su

conversión a milímetros.

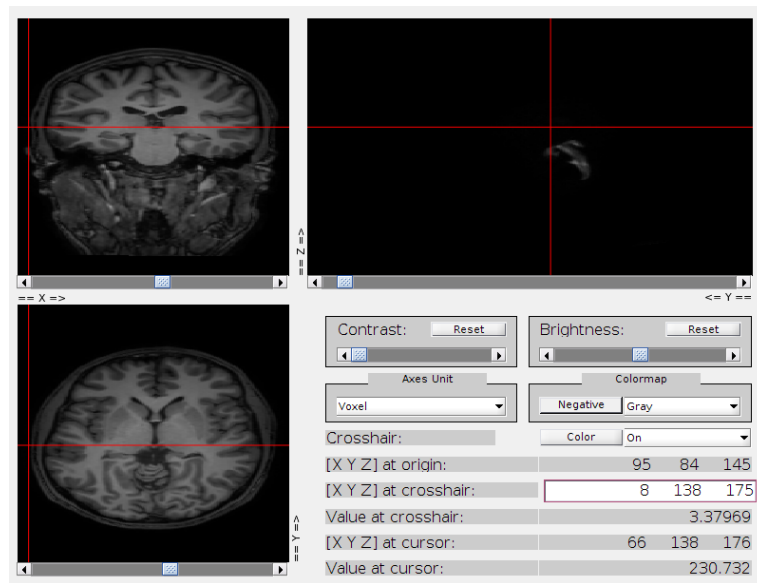


Figura 92. Coordenada en vóxel de un punto sobre la MRI del Brainsight

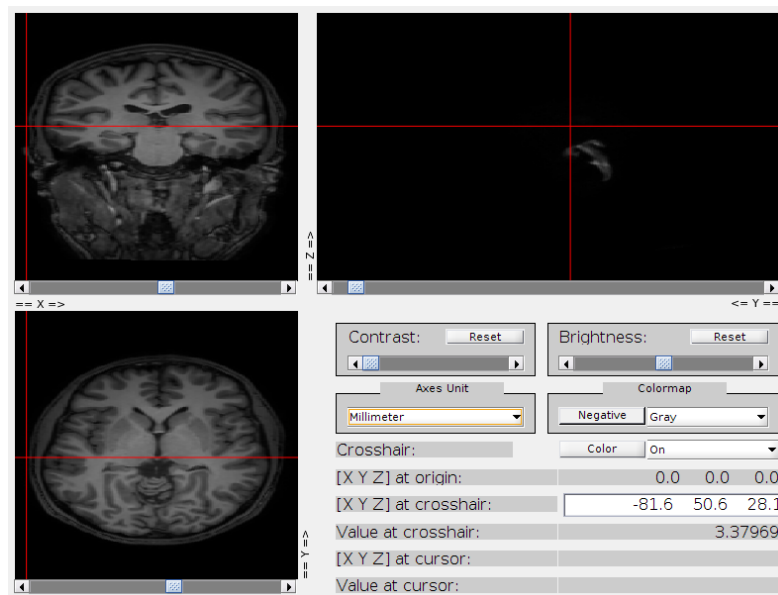


Figura 93. Coordenada en milímetros de un punto sobre la MRI del Brainsight

Por último, se puede observar que ese mismo punto ha sido localizado dentro del programa Brainsight y en la cabeza de un sujeto en el mundo real. En la interfaz que muestra Brainsight se observan varias imágenes: la MRI visualizada en diferentes vistas, una reconstrucción del sujeto en 3 dimensiones y una especie de diana con un punto rojo en el centro. En la imagen en 3 dimensiones de la cabeza (figura 94) se puede observar un puntero verde a pocos centímetros de la oreja izquierda. Este punto es el mismo que se marcó anteriormente en la superficie mallada, el cual se marcó también en la MRI. Ese puntero verde se representa como una diana de color verde en la imagen del Brainsight. Además, se observa un punto rojo, tanto en la imagen en 3 dimensiones como en el centro de la diana. Ese punto se corresponde con la zona donde está apuntando la bobina o, en este caso, el puntero que señala a la cabeza en el mundo real. Se observa también como el punto rojo está en el centro de la diana cuando el puntero señala la parte izquierda de la cabeza, por encima de la oreja (ver figura 95).

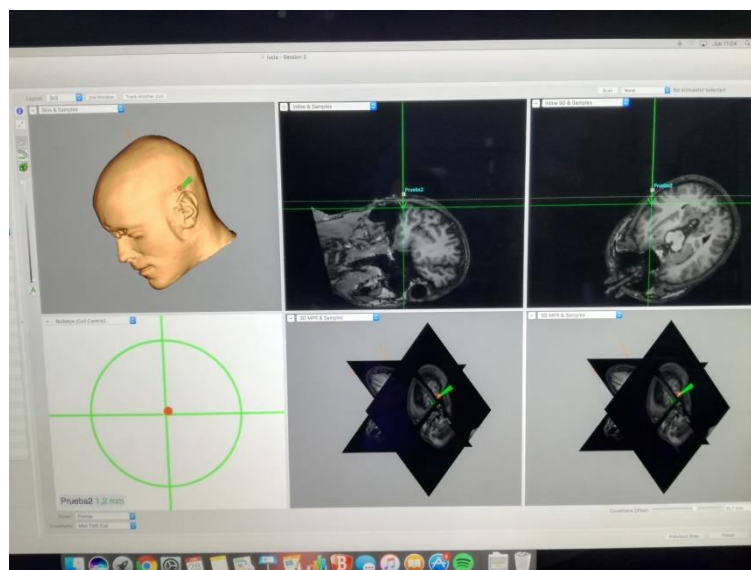


Figura 94. Interfaz del Brainsight



Figura 95. Localización de un punto en el espacio real con el software Brainsight



## 4 CONCLUSIONES

---

Los modelos de flujo de corrientes en el cerebro son muy importantes para la investigación relacionada con la estimulación eléctrica transcraneal. Para ello son necesarios los modelos computacionales del cerebro para poder realizar simulaciones. Esto requiere un procesamiento de las imágenes biomédicas [101], el cual es distinto para cada herramienta. Debido a la multitud de herramientas existentes y sus resultados en la literatura, es preciso estudiar algunas de ellas para realizar un flujo de trabajo que permita obtener un modelo computacional del cerebro lo más parecido a la realidad.

Para empezar, ha ido muy importante el estudio del formato NIFTI, junto con las demás herramientas de visualización, para comprender mejor la imagen representada, tener controlada sus coordenadas, conocer su orientación e incluso su normalización.

El estudio de diferentes programas para la segmentación de las MRIs ha sido necesario para analizar las ventajas e inconvenientes de cada uno de ellos y su posterior elección. De esta forma, se ha comprobado la ventaja que ROAST tiene sobre SPM12 al combinar las segmentaciones de T1 y T2, ofreciendo una notable mejora en la segmentación del hueso. Además, se ha descubierto la importancia de disponer una adecuada plantilla para la segmentación de las imágenes, resultando la plantilla de ROAST mucho mejor que la de SPM8. También, se ha visto que si se implementa SPM12 en ROAST, en lugar de usar SPM8, la segmentación es de más calidad, ya que ROAST con SPM8 muestra en las segmentaciones algunos vóxeles aislados de los tejidos, mientras que ROAST con SPM12 segmenta mucho mejor al no presentar esas imperfecciones en ninguno de los tejidos. En cuanto a la calidad de las segmentaciones, tras la comparativa de ROAST frente a SIMNIBS [102] decir que, SPM12 segmenta el fluido cerebroespinal y el cráneo mejor que SIMNIBS. La segmentación de SPM12 del cráneo es mucho más realista que la de SIMNIBS, ya que representa los agujeros de los ojos. Mientras que la segmentación de la materia blanca y gris realizada por SIMNIBS es de más calidad que las que realiza SPM12 [103]. Cabe destacar que SPM12 segmenta hasta el cuello mientras que SIMNIBS lo hace hasta la altura de la nariz. Sin embargo otra ventaja que presenta SIMNIBS frente a SPM12 es que segmenta también los ventrículos y cerebelo.

La importancia de la calidad de la segmentación se hace notar en la reconstrucción del mallado triangular. Tras la comparación de los métodos `binsurface` y `Vol2surf` para crear la superficie de triángulos de los tejidos, cabe destacar que, el primero de ellos realiza un proceso más lento pero más exhaustivo que el segundo.

En cuanto a los algoritmos estudiados para calcular las ubicaciones de los electrodos según un sistema internacional, se ha llegado a la conclusión de que el método utilizado por la herramienta `Mesh2EEG` es mucho más preciso que el método que usa `Metch`, por lo que se eligió para la reconstrucción de los electrodos.



## REFERENCIAS

- [1] Y. Huang, L. C. Parra y S. Haufe, «The New York Head—A precise standardized volume conductormodel for EEG source localization and tES targeting,» *NeuroImage*, vol. 140, p. 150–162, 2016.
- [2] J. Yanamadala, V.K. Rathi, S. Maliye et al., «Segmentation of the Visible Human Project® (VHP) Female Cryosection Images within MATLAB® Environment,» 23rd International Meshing Roundtable (IMR23), Oct. 1-5, 2014.
- [3] E. Dougherty y J. C. Turner, «An Object-Oriented Framework for Versatile Finite Element Based Simulations of Neurostimulation,» *Journal of Computational Medicine*, vol. 2016, 2015.
- [4] EASYCAP GmbH, EasyCap Electrode Layouts based on 10%-System, Germany.
- [5] G. B. Saturnino, SimNIBS 2.0.1 Tutorials. tDCS Simulation, 2016.
- [6] Z. Akalin Acar y S. Makeig, «Neuroelectromagnetic forward head modeling toolbox,» *Journal of Neuroscience Methods*, vol. 190, pp. 258-270, 2010.
- [7] M. Windhoff, A. Opitz y A. Thielscher, «Electric field calculations in brain stimulation based on finite elements: An optimized processing pipeline for the generation and usage of accurate individual head models,» *Human Brain Mapping*, vol. 34, pp. 923-935, 2013.
- [8] H. Behjat, Statistical Parametric Mapping of Functional MRI data Using Spectral Graph Wavelets, MSc Thesis, Linköping University, 2012.
- [9] Q. Fang y D. Boas, «iso2mesh: an image-based mesh generation toolbox,» 2009. [En línea]. Available: <http://iso2mesh.sourceforge.net> [Último acceso: 3 Julio 2018].
- [10] Q. Fang, a «iso2mesh: a “one-liner” 3D mesh generator, » 2010. [En línea]. Available: Available: <http://iso2mesh.sourceforge.net> [Último acceso: 3 Julio 2018].
- [11] T. Wagner, M. Zahn, A. Grodzinsky y A. Pascual-Leone, «Three-dimensional head model simulation of transcranial magnetic,» *IEEE Trans Bio-Med Eng*, vol. 51, p. 1586–1598, 2004.
- [12] C. C. McIntyre y T. J. Foutz, «Computational modeling of deep brain stimulation,» *Handbook of Clinical Neurology*, vol. 116, pp. 55-61, 2013.
- [13] E. Rusconi y S. Bestmann, «On tickling brains to investigate,» *Cortex*, vol. 45, p. 1021–1024, 2009.
- [14] L. A. Shepp y Y. Vardi, «Maximum likelihood reconstruction in positron emission tomography,» *IEEE Trans Med Imaging*, vol. 1, p. 113–122, 1982.
- [15] E. Ziegler, S. L. Chellappa, G. Gaggioni, J. Q. M. Ly, G. Vandewalle, E. André, C. Geuzaine y C. Phillips, «A finite-element reciprocity solution for EEG forward modeling with realistic individual head models,» *NeuroImage*, vol. 103, p. 542–551, 2014.
- [16] P. M. Bayonas, Preprocesado y Registro de Imágenes Médicas Portal con Radiografías Reconstruidas Digitalmente, Proyecto Fin de Carrera, Universidad Politécnica de Cartagena, 2010.
- [17] P. Cavaleiro Miranda, M. Lomarev y M. Hallett, «Modeling the current distribution during transcranial direct current stimulation,» *Clin Neurophysiol*, vol. 117, p. 1623–1629, 2006.
- [18] J. T. Nelson, R. A. McKinley, E. J. Golo, J. S. Warm y R. Parasuraman, «Enhancing vigilance in operators with prefrontal cortex transcranial direct current stimulation (tDCS),» *Neuroimage*, vol. 85, p. 909–917, 2014.

- [19] J. de Munck, P. van Houdt, R. M. Verdaasdonk y P. Ossenblok, «A semi-automatic method to determine electrode positions and labels from gel artifacts in EEG/fMRI-studies,» *NeuroImage*, vol. 59, p. 399–403, 2012.
- [20] S. B. Eickhoff, K. E. Stephan, H. Mohlberg, C. Grefkes, G. R. Fink, K. Amunts y K. Zilles, «A new SPM toolbox for combining probabilistic cytoarchitectonic maps and functional imaging data,» *NeuroImage*, vol. 25, p. 1325–1335, 2005.
- [21] K. de Macedo Rodrigues, E. Ben-Avi, M.-s. Choe, M. Drottar, D. D. Sliva, R. Wang, B. Fischl, P. E. Grant y L. Zöllei, «A FreeSurfer-compliant consistent manual segmentation of infant brains spanning the 0–2 year age range,» *Front Hum Neurosci*, 2015 p. 9-21, 2015.
- [22] C. Lee, Y.-J. Jung, S. J. Lee y C.-H. Im, «COMETS2: An advanced MATLAB toolbox for the numerical analysis of electric fields generated by transcranial direct current stimulation,» *Journal of Neuroscience Methods*, vol. 277, p. 56–62, 2017.
- [23] Y. Zhang, M. Brady y S. Smith, «Segmentation of brain MR images through a hidden Markov random field model and the expectation-maximization algorithm,» *IEEE Trans Med Imaging*, vol. 20, p. 45–57, 2001.
- [24] J. Schöberl, «NETGEN An advancing front 2D/3D-mesh generator,» *Comput Visual Sci*, vol. 1, p. 41–52, 1997.
- [25] C. Rorden, «MRICron Installation,» [En línea]. Available: <http://people.cas.sc.edu/rorden/mricron/install.html>. [Último acceso: 3 Julio 2018].
- [26] E. Crespo, Y. Alemán, J. Pascau, S. Reig, J. Janssen y M. Desco, «Plataforma de recepción, gestión y almacenamiento de imagen médica en formato digital DICOM,» *Congreso Anual de la Sociedad Española de Ingeniería Biomédica (CASEIB2014)*, 2014.
- [27] «MicroDicom - free DICOM viewer for Windows,» [En línea]. Available: <http://www.microdicom.com>. [Último acceso: 3 Julio 2018].
- [28] Mayo Foundation Biomedical Imaging Resource, «Analyze 7.5 specification document,» *Bio-Formats Documentation*, 2006. [En línea]. Available: <https://docs.openmicroscopy.org/bio-formats/5.9.0/formats/analyze-75.html>. [Último acceso: 3 Julio 2018]. [Último acceso: 3 Julio 2018].
- [29] F. S. Collins, W. J. Koroshetz y . N. D. Volkow, «Helping to End Addiction Over the Long-term. The Research Plan for the NIH HEAL Initiative,» *JAMA*, vol. 320, p. 129-130, 2018.
- [30] «NIfTI-1 Data Format,» [En línea]. Available: <https://nifti.nimh.nih.gov/nifti-1/> [Último acceso: 3 Julio 2018].
- [31] A. Manna, «Introducción al Procesamiento de Imágenes con Matlab, 1ª Parte,» 2016. [En línea]. Available: <https://www.dc.uba.ar/materias/t1/2016/c1/archivos/2016/introimagenes> [Último acceso: 3 Julio 2018].
- [32] B. Cox, «NIFTI toolbox,» 2007. [En línea]. Available: <https://nifti.nimh.nih.gov/pub/dist/src/niftilib/nifti1.h> [Último acceso: 3 Julio 2018].
- [33] J. Shen, «Tools for NifTI and ANALYZE image,» [En línea]. Available: <https://www.mathworks.com/matlabcentral/fileexchange/8797-tools-for-nifti-and-analyze-image> [Último acceso: 3 Julio 2018].
- [34] C. Gaser, «Structural Brain Mapping Group,» [En línea]. Available: <http://www.neuro.uni-jena.de/vbm/segmentation/> [Último acceso: 3 Julio 2018].
- [35] C. Rorden, «MRICron home page and introduction,» 2014. [En línea]. Available: <http://people.cas.sc.edu/rorden/mricron/main.html> [Último acceso: 3 Julio 2018].
- [36] E. Cuenca, «MATLAB: Introducción al procesamiento de imágenes,» 2006. [En línea]. Available: <http://informatica.uv.es/iiguia/2000/VC/tutorial.pdf> [Último acceso: 3 Julio 2018].
- [37] F. Guindos Rojas, «Tratamiento digital de imágenes con IMtdi,» 2001. [En línea]. Available: <https://w3.ual.es/~fguindos/IMtdi/Tratamiento%20Digital%20de%20Im%20genes%20con%20IMtdi.pdf> [Último acceso: 3 Julio 2018].



- [38] Rjpatruno, «Anatomical-Processing,» [En línea]. Available: <http://web.stanford.edu/group/vista/cgi-bin/wiki/index.php/Anatomical-Processing>. [Último acceso: 3 Julio 2018].
- [39] M. Aziz, *Is it left or is it right?*, 2009.
- [40] A. Gripschofer y S. Westerheide, «Matlab toolbox for pet / ct image segmentation with the Chan-Vese model,» 2008. [En línea]. Available: <https://www.unimuenster.de/AMM/num/Vorlesungen/MedizinUndMathematik/PETCT/documentation.pdf>. [Último acceso: 3 Julio 2018].
- [41] «About the MNI space(s),» [En línea]. Available: <http://www.lead-dbs.org/?p=1241> [Último acceso: 3 Julio 2018].
- [42] «Parkinson's Progression Markers Initiative,» [En línea]. Available: <https://www.ppmi-info.org/about-ppmi/who-we-are/>. [Último acceso: 3 Julio 2018].
- [43] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald y W. Stuetzle, «Mesh Optimization,» *Proc 20th Ann Conf Computer Graphics Interactive Techniques*, p. 19-26, 1993.
- [44] «Segmentación (procesamiento de imágenes),» [En línea]. Available: [https://es.wikipedia.org/wiki/Segmentaci%C3%B3n\\_\(procesamiento\\_de\\_im%C3%A1genes\)](https://es.wikipedia.org/wiki/Segmentaci%C3%B3n_(procesamiento_de_im%C3%A1genes)). [Último acceso: 5 Junio 2018].
- [45] T. C. Pataky, J. Vanrenterghem y M. Robinson, «Statistical parametric mapping (SPM): theory, software and future directions,» [En línea]. Available: [http://www.spm1d.org/\\_downloads/Slides-ISP2017.pdf](http://www.spm1d.org/_downloads/Slides-ISP2017.pdf) [Último acceso: 3 Julio 2018].
- [46] D. Güllmar, J. Haueisen y J. R. Reichenbach, «Influence of anisotropic electrical conductivity in white matter tissue on the EEG/MEG forward and inverse solution. A high-resolution whole head simulation study,» *Neuroimage*, vol. 51, p. 145–163, 2010.
- [47] M. G. Gutiérrez Romero, «Diseño e implementación de una herramienta computacional dinámica para mejorar las habilidades de memoria de adultos mayores, y comprobar su eficacia por medio de la detección de actividad cerebral utilizando imágenes de resonancia magnética,» *Proyecto Fin de Carrera*, Universidad Católica de Loja (Ecuador), 2017.
- [48] A. C. Evans, A. L. Janke, D. L. Collins y S. Baillet, «Brain templates and atlases,» *NeuroImage*, vol. 62, p. 911–922, 2012.
- [49] J. D. Nielsen, K. H. Madsen, O. Puonti, H. R. Siebner, C. Bauer, C. Göbel Madsen, G. B. Saturnino y A. Thielscher, «Automatic skull segmentation from MR images for realistic volume conductor models of the head: Assessment of the state-of-the-art,» *NeuroImage*, 2017.
- [50] J. Muschelli, «Wrapper Functions for 'SPM' (Statistical Parametric Mapping),» 2018. [En línea]. Available: <http://johnmuschelli.com/spm12r/> [Último acceso: 3 Julio 2018].
- [51] O. Chancay, T. Haro, M. Yapur, R. Alvarado, M. Pastor y F. Loayza, «Nuevo Biomarcador en la Enfermedad de Parkinson Mediante el Análisis y Cuantificación de Lesiones Cerebrales en Secuencias Flair Obtenidas por Resonancia Magnética (ACL-Tool),» *Revista Tecnológica ESPOL*, vol. 28, pp. 548-557, 2015.
- [52] B. Dogdas, D. W. Shattuck y R. M. Leahy, «Segmentation of Skull and Scalp in 3-D Human MRI Using Mathematical Morphology,» *Human Brain Mapping*, vol. 26, p. 273–285, 2005.
- [53] «SPM12 Release Notes,» London WC1N 3BG, UK. [En línea]. Available: [https://www.fil.ion.ucl.ac.uk/spm/software/spm12/SPM12\\_Release\\_Notes.pdf](https://www.fil.ion.ucl.ac.uk/spm/software/spm12/SPM12_Release_Notes.pdf). [Último acceso: 3 Julio 2018].
- [54] Y. Huang y L. C. Parra, «Fully Automated Whole-Head Segmentation with Improved Smoothness and Continuity, with Theory Reviewed,» *PLoS One*, vol. 10, 2015.
- [55] Z. Akalin Acar y S. Makeig, «Neuroelectromagnetic Forward Head Modeling Toolbox,» *J Neurosci Methods*, vol. 190, p. 258–270, 2010.
- [56] R. Nordenskjöld, F. Malmberg, E.-M. Larsson, A. Simmons, H. Ahlström, L. Johansson y J. Kullberg,

- «Intracranial volume normalization methods: Considerations when investigating gender differences in regional brain volume,» *Neuroimaging*, vol. 231, p. 227–235, 2015.
- [57] V. Fonov, A. C. Evans, K. Botteron, C. R. Almli, R. C. McKinstry y D. L. Collins, «Unbiased Average Age-Appropriate Atlases for Pediatric Studies,» *NIH Public Access*, vol. 54, p. 313–32, 2011.
- [58] J. Ashburner, *VBM Tutorial*, 2010.
- [59] Y. Huang, A. Datta, M. Bikson y L. C. Parra, «ROAST: an open-source, fully-automated, Realistic vOlumetric-Approach-based Simulator for TES»,
- [60] R. P. Woods, S. T. Grafton, C. J. Holmes, S. R. Cherry y J. C. Mazziotta, «Automated image registration: I. General methods and intrasubject, intramodality validation,» *Journal of Computer Assisted Tomography*, vol. 22, pp. 139–152, 1998.
- [61] C. Xiaohua, M. Brady y D. Rueckert, «Simultaneous Segmentation and Registration for Medical Image,» *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2004.
- [62] S. Hu, Y. Lai, P. A. Valdes-Sosa y M. L. Bringas-Vega, «How do reference montage and electrodes setup affect the measured scalp EEG potentials?,» *Journal of Neural Engineering*, 2018.
- [63] J. E. Richards, C. Boswell, M. Stevens y J. M. C. Vendemia, «Evaluating Methods for Constructing Average High-Density Electrode Positions,» *Brain Topogr*, vol. 28, p. 70–86, 2015.
- [64] S. M. Smith, «Fast robust automated brain extraction. *Human Brain Mapping*,» *Human Brain Mapping*, vol. 17, 2002, p. 143–155.
- [65] «FreeSurfer,» [En línea]. Available: <https://surfer.nmr.mgh.harvard.edu/fswiki>. [Último acceso: 3 Julio 2018].
- [66] M. Windhoff, «How to create a head model,» [En línea]. Available: <http://simnibs.de/version2/documentation>. [Último acceso: 3 Julio 2018].
- [67] P. Alliez, *Mesh Optimization*, 2018.
- [68] A. M. Dale, B. Fischl y M. I. Sereno, «Cortical surface-based analysis. I. Segmentation and surface reconstruction,» *Neuroimage*, vol. 9, p. 179–94, 1999.
- [69] Q. Fang, «Iso2mesh,» [En línea]. Available: <http://iso2mesh.sourceforge.net/cgi-bin/index.cgi>. [Último acceso: 3 Julio 2018].
- [70] Q. Fang y D. A. Boas, «Tetrahedral mesh generation from volumetric binary and grayscale images,» *Proc 2009 IEEE Int Symp Biomed Imaging: From Nano to Macro*, 2009.
- [71] K. Van Leemput, F. Maes, D. Vandermeulen y P. Suetens, «A Statistical Framework for Partial Volume Segmentation,» *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 204–212, 2001.
- [72] R. Oostenveld y P. Praamstra, «The five percent electrode system for high-resolution EEG and ERP measurements,» *Clinical Neurophysiology*, vol. 112, pp. 713–719, 2001.
- [73] M. Hart. [En línea]. Available: <http://www.betereidingen.nl>. [Último acceso: 3 Julio 2018].
- [74] P. Hea y J. R. Estepb, «A practical method for quickly determining electrode positions in high-density EEG studies,» *Neuroscience Letters*, vol. 541, p. 73–76, 2013.
- [75] R. Oostenveld y P. Praamstrac, «The five percent electrode system for high-resolution EEG and ERP measurements,» *Clinical Neurophysiology*, vol. 112, pp. 713–719, 2001.
- [76] K. M. Benhadida, V. Brauna, «Spatial localization of EEG electrodes,» *Clinical Neurophysiology*, vol. 37, p. 97–102, 2007.
- [77] L. Koessler, L. Maillard, A. Benhadid, et al., «Automated cortical projection of EEG sensors: Anatomical correlation via the international 10–10 system,» *NeuroImage*, vol. 46, p. 64–72, 2009.
- [78] S.-S. Yoo, C. Guttman, J. R. Ives, L. Panuch, R. K. D. Schomer y F. Jolesz, «3D Localization of surface 10–20 EEG electrodes on high resolution anatomical MR images,» *Electroencephalography and clinical*

- Neurophysiology, vol. 102, pp. 335-339, 1997.
- [79] A. Datta, V. Bansal, J. Diaz, J. Patel, D. Reato y M. Bikson, «Gyri-precise head model of transcranial direct current stimulation: Improved spatial focality using a ring electrode versus conventional rectangular pad,» *Brain Stimulation*, vol. 2, n° 7, p. 201, 2009.
  - [80] Y. H. A. Datta, M. Bikson y L. C. Parra, «Realistic vOlumetric-Approach to Simulate Transcranial Electric Stimulation – ROAST – a fully automated open-source pipeline,» New York, 2017.
  - [81] K. Kowkabzadeh, «Evaluations of Tissue Segmentation of brain MR Images,» MSc Thesis, Chalmers University of Technology, 2010.
  - [82] M. Dannhauer, . D. Brooks, D. Tucker y R. MacLeod, «A pipeline for the simulation of transcranial direct current stimulation for realistic human head models using SCIRun/BioMesh3D,» *Proc Ann Int Conf IEEE Engineering in Medicine and Biology Society (EMBC)*, 2012.
  - [83] A. Thielscher, A. Antunes y G. B. Saturnino, «Field modeling for transcranial magnetic stimulation: a useful tool to understand the physiological effects of TMS?,» *Proc 37th Ann Int Conf IEEE Engineering in Medicine and Biology Society*, August 25-29, 2015, Milano Conference Center, Milano, Italy, 2015.
  - [84] J. Shen, Frequently Asked Questions for NIfTI and ANALYZE tools.
  - [85] Y. Huang, J. P. Dmochowski, Y. Su, A. Datta, C. Rorden y L. C. Parra, «Automated MRI segmentation for individualized modeling of current flow in the human head,» *Journal of Neural Engineering*, vol. 10, p. 13, 2013.
  - [86] W. Wells, W. Grimson, R. Kikinis y F. Jolesz, «Adaptive segmentation of MRI data,» *IEEE Trans Med Imaging*, vol. 15, p. 429–442, 1996.
  - [87] J. Ashburner y K. , «Unified segmentation,» *NeuroImage*, vol. 26, p. 839– 851, 2005.
  - [88] G. Ridgway, *Voxel-Based Morphometry*, Oxford & UCL, 2016.
  - [89] M. W. Chau y A. , «The Talairach coordinate of a point in the MNI space: how to interpret it,» *NeuroImage*, vol. 25, p. 408– 416, 2005.
  - [90] R. P. Woods, «Characterizing volume and surface deformations in an atlas framework: theory, applications, and implementation,» *NeuroImage*, vol. 18, p. 769–788, 2003.
  - [91] C. Jamin, P. Alliez, M. Yvinec y J.-D. Boissonnat, «CGALmesh: a Generic Framework for Delaunay Mesh,» *ACM Transactions on Mathematical Software*, Vol. 41, 2014.
  - [92] L. Rineau, S. Tayeb y M. Yvinec, «CGAL 4.12 - Manual,» 2009. [En línea]. Available: <https://doc.cgal.org/latest/Manual/packages.html>. [Último acceso: 25 Abril 2018].
  - [93] J. Le, M. Lu, E. Pellouchoud y A. Gevins, «A rapid method for determining standard 10/10 electrode positions for high resolution EEG studies,» *Electroencephalography and clinical Neurophysiology*, vol. 106, p. 554–558, 1998.
  - [94] P. Giacometti, K. L. Perdue y S. G. Diamond, «Algorithm to find high density EEG scalp coordinates and analysis of their correspondence to structural and functional regions of the brain,» *NIH Public Acce*, vol. 229, p. 84–96, 2014.
  - [95] D. T. a. I. D. Valer Jurcak1, «10/20, 10/10, and 10/5 systems revisited: Their validity as relative head-surface-based positioning systems,» *NeuroImage*, vol. 34, p. 1600–1611, 2007.
  - [96] N. Parra-Bolaños, «Impacto de las técnicas de neuroimagen en las ciencias sociales,» *Rev. Chil. Neuropsicol*, vol. 10, pp. 31-37, 2015.
  - [97] J. L. Lancaster, D. Tordesillas-Gutierrez, M. Martinez, F. Salinas, A. Evans, K. Zilles, J. C. Mazziotta y P. T. Fox, «Bias Between MNI and Talairach Coordinates Analyzed Using the ICBM-152 Brain Template,» *Human Brain Mapping*, vol. 28, p. 1194–1205, 2007.
  - [98] J. Wu, G. H. Ngo, D. Greve, J. Li, T. He, B. Fischl, S. B. Eickhoff y T. Yeo, *Accurate Nonlinear Mapping between MNI Volumetric and FreeSurfer Surface Coordinate Systems*, Singapore, 2018.
  - [99] W. M. Wells, P. Viola, H. Atsumi, S. Nakajima y R. Kikinis, «Multi-modal volume registration by

- maximization of mutual information,» *Med Image Anal*, vol. 1, pp. 35-51, 1996.
- [100] Y. Huang, A. A. Liu, B. Lafon, D. Friedman, M. Dayan, X. Wang, M. Bikson, W. K. Doyle, O. Devinsky y L. C. Parra, «Measurements and models of electric fields in the in vivo human brain during transcranial electric stimulation,» *eLife*, 2017.
- [101] M. Ida Iacono, E. Neufeld, E. Akinagbe, K. Bower, J. Wolf, I. Vogiatzis Oikonomidis, D. Sharma, B. Lloyd, B. J. Wilm, M. Wyss, K. P. Pruessmann, A. Jakab, N. Makris, E. D. Cohen, N. Kuster, W. Kainz y L. M. Angelone, «MIDA: A Multimodal Imaging-Based Detailed Anatomical Model of the Human Head and Neck,» *PLoS One*, vol. 10, 2015.
- [102] G. Saturnino, J. Stelzer y A. Thielscher, «SimNIBS: A versatile toolbox for simulating fields generated by transcranial brain stimulation,» *Proc 21st Annual Meeting of the Organization for Human Brain Mapping (OHBM 2015)*, 2015.
- [103] I. Malone, K. Leung, C. Shona y J. Barnes, «Accurate automatic estimation of total intracranial volume: A nuisance variable with less nuisance,» *NeuroImage*, vol. 104, pp. 366-372, 2015.
- [106] A. Datta, M. Bikson y F. Fregni, «Transcranial direct current stimulation in patients with skull defects and skull plates: High-resolution computational FEM study of factors altering cortical current flow,» *Neuroimage*, vol. 52, p. 1268–1278, 2010.
- [107] J. Pascau, P. Rojo, A. Santos, M. A. Pozo y M. Desco, «Localización espacial de dipolos de electroencefalografía en imágenes de resonancia magnética,» *Congreso Anual de la Sociedad Española de Ingeniería Biomédica (CASEIB2014)*, 2014.
- [108] R. Kempe, «Sampling pad-electrode configurations with a high density electrode array for transcranial direct current stimulation,» *Journal of Neural Engineering*, vol. 11, 2014.
- [109] M. Marino, Q. Liu, S. Brem, N. Wenderoth y D. Mantini, «Automated detection and labeling of high-density EEG electrodes from structural MR images,» *Journal of Neural Engineering*, vol. 13, 2016.
- [110] A. Sack, R. Cohen Kadosh, et al., «Optimizing functional accuracy of TMS in cognitive,» *J Cogn Neurosci*, vol. 21, p. 207–22, 2009.
- [111] C. Geuzaine y J. Remacle, «A three-dimensional finite element mesh generator with built-in pre- and post-processing facilities,» *Int J Numer Meth Eng*, vol. 79, p. 1309–1331, 2009.